Semantic Spaces for Video Analysis of Behaviour

Xun Xu

Submitted in partial fulfilment of the requirement for the degree of *Doctor of Philosophy*

School of Electronic Engineering and Computer Science

Queen Mary University of London

17 November 2016

Semantic Spaces for Video Analysis of Behaviour

Xun Xu

Abstract

There are ever growing interests from the computer vision community into human behaviour analysis based on visual sensors. These interests generally include: (1) behaviour recognition - given a video clip or specific spatio-temporal volume of interest discriminate it into one or more of a set of pre-defined categories; (2) behaviour retrieval - given a video or textual description as query, search for video clips with related behaviour; (3) behaviour summarisation - given a number of video clips, summarise out representative and distinct behaviours. Although countless efforts have been dedicated into problems mentioned above, few works have attempted to analyse human behaviours in a semantic space. In this thesis, we define semantic spaces as a collection of high-dimensional Euclidean space in which semantic meaningful events, e.g. individual word, phrase and visual event, can be represented as vectors or distributions which are referred to as semantic representations. With the semantic space, semantic texts, visual events can be quantitatively compared by inner product, distance and divergence. The introduction of semantic spaces can bring lots of benefits for visual analysis. For example, discovering semantic representations for visual data can facilitate semantic meaningful video summarisation, retrieval and anomaly detection. Semantic space can also seamlessly bridge categories and datasets which are conventionally treated independent. This has encouraged the sharing of data and knowledge across categories and even datasets to improve recognition performance and reduce labelling effort. Moreover, semantic space has the ability to generalise learned model beyond known classes which is usually referred to as zero-shot learning. Nevertheless, discovering such a semantic space is non-trivial due to (1) semantic space is hard to define manually. Humans always have a good sense of specifying the semantic relatedness between visual and textual instances. But a measurable and finite semantic space can be difficult to construct with limited manual supervision. As a result, constructing semantic space from data is adopted to learn in an unsupervised manner; (2) It is hard to build a universal semantic space, i.e. this space is always contextual dependent. So it is important to build semantic space upon selected data such that it is always meaningful within the context. Even with a well constructed semantic space, challenges are still present including; (3) how to represent visual instances in the semantic space; and (4) how to mitigate the misalignment of visual feature and semantic spaces across categories and even datasets when knowledge/data are generalised. This thesis tackles the above challenges by exploiting data from different sources and building contextual semantic space with which data and knowledge can be transferred and shared to facilitate the general video behaviour analysis.

To demonstrate the efficacy of semantic space for behaviour analysis, we focus on studying real world problems including surveillance behaviour analysis, zero-shot human action recognition and zero-shot crowd behaviour recognition with techniques specifically tailored for the nature of each problem.

Firstly, for video surveillances scenes, we propose to discover semantic representations from the visual data in an unsupervised manner. This is due to the largely availability of unlabelled

visual data in surveillance systems. By representing visual instances in the semantic space, data and annotations can be generalised to new events and even new surveillance scenes. Specifically, to detect abnormal events this thesis studies a geometrical alignment between semantic representation of events across scenes. Semantic actions can be thus transferred to new scenes and abnormal events can be detected in an unsupervised way. To model multiple surveillance scenes simultaneously, we show how to learn a shared semantic representation across a group of semantic related scenes through a multi-layer clustering of scenes. With multi-scene modelling we show how to improve surveillance tasks including scene activity profiling/understanding, cross-scene query-by-example, behaviour classification, and video summarisation.

Secondly, to avoid extremely costly and ambiguous video annotating, we investigate how to generalise recognition models learned from known categories to novel ones, which is often termed as zero-shot learning. To exploit the limited human supervision, e.g. category names, we construct the semantic space via a word-vector representation trained on large textual corpus in an unsupervised manner. Representation of visual instance in semantic space is obtained by learning a visual-to-semantic mapping. We notice that blindly applying the mapping learned from known categories to novel categories can cause bias and deteriorating the performance which is termed as domain shift. To solve this problem we employed techniques including semi-supervised learning, self-training, hubness correction, multi-task learning and domain adaptation. All these methods in combine achieve state-of-the-art performance in zero-shot human action task.

In the last, we study the possibility to re-use known and manually labelled semantic crowd attributes to recognise rare and unknown crowd behaviours. This task is termed as zero-shot crowd behaviours recognition. Crucially we point out that given the multi-labelled nature of semantic crowd attributes, zero-shot recognition can be improved by exploiting the co-occurrence between attributes.

To summarise, this thesis studies methods for analysing video behaviours and demonstrates that exploring semantic spaces for video analysis is advantageous and more importantly enables multi-scene analysis and zero-shot learning beyond conventional learning strategies.

Submitted in partial fulfilment of the requirement for the degree of *Doctor of Philosophy*

School of Electronic Engineering and Computer Science

Queen Mary University of London

17 November 2016

Declaration

I hereby declare that this thesis has not been submitted, either in the same or different form, to this or any other university for a degree. Some parts of the work have been published or submitted for review.

- **Chapter 3**
  - X. Xu, S. Gong and T. Hospedales, "Cross-domain traffic scene understanding by motion model transfer," In Proceedings of the 4th ACM/IEEE international workshop on Analysis and retrieval of tracked events and motion in imagery stream (ARTEMIS), pp.77-86, 2013
- **Chapter 4**
  - X. Xu, T. Hospedales and S. Gong. "Discovery of Shared Semantic Spaces for Multi-Scene Video Query and Summarization," in IEEE Transactions on Circuits and Systems for Video Technology (CSVT), vol.PP, no.99, pp 1-1, 2016
- **Chapter 5**
  - X. Xu, T. Hospedales and S. Gong, "Semantic embedding space for zero-shot action recognition," in IEEE International Conference on Image Processing (ICIP), pp. 63-67, 2015
  - X. Xu, T. Hospedales and S. Gong, "Zero-Shot Action Recognition by Word-Vector Embedding," submitted to International Journal of Computer Vision (IJCV), 2016
- **Chapter 6**
  - X. Xu, T. Hospedales and S. Gong, "Multi-Task Zero-Shot Action Recognition with Prioritised Data Augmentation," European Conference on Computer Vision (ECCV), 2016
- **Chapter 7**

– X. Xu, S. Gong and T. Hospedales, “Zero-Shot Crowd Behaviour Recognition,” In Murino, Shah, Cristani, Savarese (Eds.), Group and Crowd Behaviour Understanding in Computer Vision, Elsevier, 2016

Acknowledgements

In the first place, I would like to thank my supervisor Prof. Shaogang Gong for his perpetual patience, continuous encouragement and deep trust in me. I could not even have the opportunity to start my research career in Computer Vision without Prof. Gong's encouragement and supervision. Then I am deeply grateful for the invaluable advise and unreserved support from Dr. Timothy Hospedales. I can't imagine how could I overcome the difficulties without Dr. Hospedales's experience and dedication. I also greatly appreciate my second supervisor Dr. Tao Xiang's rigorous and insightful comments in every step of my PhD study. Without their hard work and participation I can hardly achieve the goals for a successful PhD nor develop the skills to carry on independent research.

My sincere gratitude goes to all computer vision group members for the help, friendship and support, in particular Dr. Yi-Zhe Song, Dr. Miles Hansard, Dr. Ke Chen, Dr. Xiatian (Eddy) Zhu, Dr. Yanwei Fu, Dr. Zhiyuan Shi, Dr. Ryan Layne, Dr. Yi Li, Wenzhao Li, Dr. Brais Cancela, Dr. Taiqing Wang, Hanxiao Wang, Yongxin Yang, Dr. Zhenyong (Ian) Fu, Elyor Kodirov, Li Zhang, Jingya Wang, Qian Yu, Kunkun Pang, Tanmoy Mukherjee, Dr. Yaowei Wang, Xiangyu Kong, Lu Tian, Shuxin Ouyang, Qi Dong, Xiaobin Chang, Chenyang Zhao and Jifei Song. I appreciate the friendship I developed with Dr. Guangwei Jiang, Dr. Yun Zhou, Dr. Peng Lin, Siying Wang and Dr. Heng Yang who have given me support and encouragement during my 4 years study. I also express deeply appreciation to Prof. Yusheng Liu and Prof. Kai Liu who encouraged me to pursue my PhD study.

At last, I would like to thank my parents and grandparents for their perpetual love, support and understanding. Most importantly, I have to thank my girlfriend Jiawen Cheng who is the best supporter of my study and life.

Contents

List of Figures

Chapter 1

Introduction

The interest into automatic behaviour understanding from a computer vision perspective of view has a long history. This endeavour has in particular developed in recent decades due to the proliferation of visual sensors like CCTV cameras, consumer video cameras and more recently robotics and human-computer interaction [1]. As a consequence unprecedented amounts of data are generated everyday from different sources which can easily overwhelm human viewers either as surveillance operators or data annotators. Despite the range of visual content to be recorded, people are particularly interested in human behaviours because of the central status in social interaction and communication. As a result, automated approaches into behaviour analysis with minimal human supervision in different contexts are desperately needed.

1.1 Video Behaviour Analysis

The need to automated visual perception, especially understanding video behaviours has been intensified by the recent proliferation of surveillance videos and online social media videos, e.g. YouTube and Vimeo. In surveillance videos, huge amount of video data is generated from every CCTV camera 24 hours 7 days a week. Moreover, it is estimated there are 4-5.9 million CCTV cameras in UK alone thus any attempt to allocate human operators to monitor even a fraction of these cameras on live would fail. As humans are always of the central interest in surveillance systems, it is highly desirable to employ an automated system to analyse human or vehicle behaviours, e.g. 'two persons fighting', 'a group of people protesting on the street' or 'vehicles violating traffic rules'. All these demands raise new challenges for conventional

surveillance systems. Apart from the surveillance video data, online video depositories, e.g. YouTube and Vimeo, are becoming popular due to the encouraging of sharing video contents on the internet. Classifying or tagging user uploaded videos into categories, e.g. 'riding bike' and 'playing basketball', for the benefit of content retrieval, filtering and recommendation has thus risen as a new problem. This demand has even encouraged an annual contest on multi-media event detection - TRECVID MED [2]. As most of these user created videos record daily activities content-based video retrieval would be largely improved should automated recognition of these behaviours be available. Automated behaviour analysis/recognition could also play an important role in human-computer interface and patient monitoring systems [3] where recognising human behaviours is a pre-requisite.

In the following sections, we first introduce a taxonomy of video behaviours and then define the specific tasks addressed in this thesis. Finally we analyse the challenges and solutions at the end of this chapter.

1.1.1 Taxonomy of Video Behaviours

In the first place, we briefly introduce the taxonomy of behaviour following Gong et al.[1] and Aggarwal et al.[3]. The behaviour here specifically refers to human behaviour and behaviours of objects operated by humans, e.g. vehicles. A categorisation of human behaviours is given in Table 1.1. **Atomic actions** are usually the most basic components in human behaviours. While it may have various presentations under different contexts. In human action recognition, this basic component can be defined as most basic body movement e.g. 'raise left arm' or 'stretch leg'. Another presentation of atomic actions in traffic analysis could be a tiny bit of foreground motion caused by vehicles or pedestrians. **Actions** are usually composed of atomic actions in a short period with a semantic meaningful purpose and usually conducted by an individual person or object, e.g. 'bay crawling', 'ride a bike' and'skiing' (see Figure 1.1 (a)). In this thesis we treat actions are the basic instances /samples in the study of video behaviour analysis because of its important role in video content retrieval, surveillance and human-computer interaction. **Activities** or **Interactions** are defined as the interactions of multiple actions over a relatively longer period of time. Activities/Interactions are visually and semantically more complex and richer than simple actions due to the involvement with multiple actions in a spatio-temporal order. The spatio-temporal order often carries very important information for surveillance purposes. In this thesis, we specifically study activities/interactions in the traffic surveillance context, e.g.

activity is defined as a typical traffic junction cycle as seen in Figure 1.1 (b). In such a traffic cycle activity, individual actions are defined as the motion of semantic coherent objects, e.g. 'a few cars travelling horizontally'. **Group Activities** or **Crowds** are defined at a higher level than activities by conceptual groups composed of multiple person and objects with collective aims of actions, e.g. 'crowd crossing street' and 'people marching on street' (Figure 1.1 (c)). In particular, crowd behaviour differ from activities/interactions in that individual behaviours are hard to distinguish from the majority. As a result crowd behaviours are analysed in a collective way rather individually.

Table 1.1: Taxonomy of video behaviours

Types	Descriptions
Atomic Actions	Instantaneous atomic entities upon which an action is formed, e.g. 'raise left arm'
Actions	A sequence of atomic actions that accomplish a purpose, e.g. 'baby crawling'
Activities/Interactions	Composed of sequences of actions over space and time. Interaction with multiple objects are common in activities/interactions, e.g. traffic junction cycle
Group Activities/Crowd	Composed of multiple persons and objects with collect aims. Crowd behaviours are usually analysed collectively rather than individually, e.g. 'crowd crossing street'.

1.1.2 General Tasks

In this thesis, we would generally focus on the analysis of human actions, traffic activities/interactions and crowd behaviours. More specifically, we are particularly interested in **Action/Activity/Crowd Recognition**. This is a process of categorizing video clips of interest into a set of know classes. Recognition can play an important role in various application contexts including visual surveillance, human computer interaction, etc. For visual surveillance we would like to categorise traffic or human behaviours into normal and abnormal, namely anomaly detection [11, 8]. This is essential to the active surveillance system which not only passively provides a record of history but also actively gives alert or even prevents hazard from happening. For human action recognition and crowd behaviour analysis, we wish to categorise action/crowd videos into one or more predefine classes for the purpose of automatic video tagging [4], video retrieval [6, 2] and crowd profiling [10].

(a) Example human actions [4, 5, 6, 7]

(b) Example human/vehicle interactions [8, 9]

(c) Example group behaviours [10]

Figure 1.1: Exemplars of typical actions, interactions and group behaviours. (a) Typical individual actions focused on human behaviours. (b) Interactions between vehicles captured by surveillance cameras. (c) Group behaviours of crowd people mostly captured by surveillance cameras.

Another interesting task in behaviour analysis is **Video Content Retrieval**. A general pipeline of query by example is by providing a visual or textual query, relevant visual content are retrieved from a large database. For surveillance systems, huge amount of visual data are generated everyday from different sources. It is impractical to allow humans to exhaustively monitor or watch every single piece of video clip to discover the wanted content. A desirable scenario is to retrieve relevant video content when an example behaviour is provided. For online video repository, retrieval is of even more interest where user is expected to provide a textual query and the system is able to return the video contents based on the semantic relevance to query rather than tag matching. Therefore an automated approach into video content retrieval based on the understanding of behaviours are of great practical value.

Video Summarisation is also a important topic especially in visual surveillance and addressed in this thesis. It assumes that most typical behaviours in surveillance context are repetitive. Thus it would greatly reduce the burden of human operators if an abstract of representative and distinct behaviours can be selected from a prolonged video.

1.2 Semantic Spaces for Video Behaviour Analysis

In this section, we briefly introduce how semantic spaces can be integrated with video behaviour analysis with focus on three widely appreciated real-world problems. Different semantic spaces are introduced as well for the need of specific problems.

In the conventional automated video behaviour analysis, people have developed supervised learning approaches to automated behaviour recognition [12, 13, 14, 15, 10]. Without loss of generality, these supervised approaches follow a same pipeline by firstly extracting visual features from video data. Then classifiers are trained for each individual class of behaviour. This conventional pipeline was proved to be effective under the assumption of sufficient labelled data for each category which, however, no longer holds in many emerging behaviour analysis scenarios. For example, the need for increasing coverage and finer classification of human actions results in more diverse and complex action categories. As an evidence, action recognition dataset size and number of categories has experienced constant growth since the classic KTH Dataset [12] (6 classes, 2004): Weizmann Dataset [16] (9 classes, 2005), Hollywood2 Dataset [17] (12 classes, 2009), Olympic Sports Dataset [7] (16 classes, 2010), HMDB51 [4] (51 classes, 2011) and UCF101 [5] (101 classes, 2012). The growing number and complexity of actions result

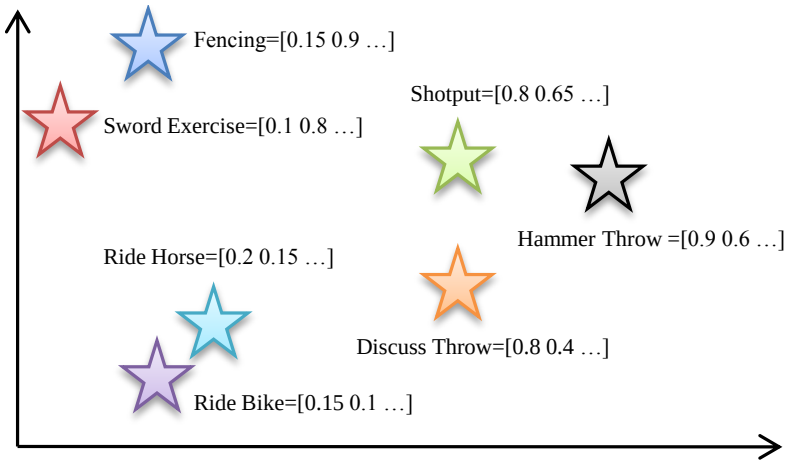


Figure 1.2: Examples of textual words in word-vector semantic space. Semantic related words and phrases are closer to each other in this space.

in: (1) enormous human effort is required to collect and label large quantities of video data for learning. Moreover, compared to image annotation, obtaining each annotated action clip is more costly as it typically requires some level of spatio-temporal segmentation from the annotator; and (2) the growing number of categories eventually begins to pose ontological difficulty, about how to structure and define distinct action categories as they grow more fine-grained and inter-related [18]. Therefore, manually labelling enough training videos for every emerging category is not scalable to large video database.

To ameliorate these issues people have proposed to exploit the semantic spaces to enable sharing information across categories [19, 20, 21, 22, 23] and crucially by allowing recognisers for novel categories to be constructed based on a human description of the action, without any labelled training samples for that particular type of action. This way of learning is often referred to as 'zero-shot learning' (ZSL) [19] in the literature.

One of the most widely exploited semantic space is by learning distributed representations of words in a vector space. The distributed semantic vector has been the focus of natural language processing community [24, 25] and can be re-used for visual behaviour analysis at zero cost. For instance, the word2vec [24] model which is trained on a large text corpus brings a by-product of words represented as real-valued vectors. Because of the large training data, the word2vec space captured the semantic relatedness between each word. Simply put, synonyms or semantic related words and phrases are closer to each other in this space, as illustrated in Figure 1.2. 'Fencing' and 'Sword Exercise' are semantically highly related, thus tend to stay closer to each other than 'Ride Horse' or 'Shotput'.

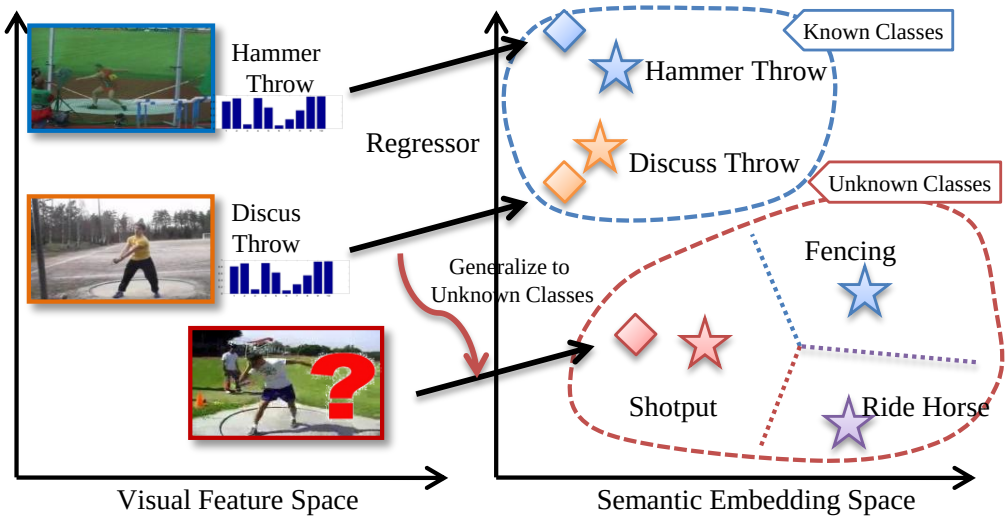


Figure 1.3: A general pipeline for word-vector based ZSL.

The most common approaches towards ZSL [26, 27, 28, 29] are to exploit the word-vector semantic space. We illustrate a general pipeline for ZSL with semantic word-vector in Figure 1.3. Regressors, are trained on the known dataset, e.g. 'Hammer Throw' and 'Discuss Throw', to map low-level visual features into this semantic embedding space. Zero-shot recognition is subsequently performed by mapping novel/unknown category visual instances to the semantic embedding space via the learned regression, and classifying these according to the vector representation of novel class names, .e.g. 'Shotput','Fencing' and 'Ride Horse' in the semantic space. Such a semantic space is often referred to as semantic embedding space as a new embedding space is created between the original visual feature space and the discrete high-level category space.

Learning semantic representations from visual data is an alternative to learning from text and can benefit visual behaviour analysis as well. In typical surveillance systems, people are often interested in some of the key tasks including: (1) behaviour profiling / scene understanding to reveal what are the typical activities and behaviours in the surveilled space [30, 9, 31, 32, 33]; (2) behaviour query by example, allowing the operator to search for similar occurrences to a specified example behaviour [30]; (3) supervised learning to classify/annotate activities or behaviours if events of interest are annotated in a training dataset [9]; (4) summarisation to give an operator a semantic overview of a long video in a short period of time [34]; and (5) anomaly detection to highlight to an operator the most unusual events in a recording period [30, 9, 31]. Instead of

modelling directly on pixel level motion features, e.g. trajecotory and optical flow, a more robust approach to activity representation is by grouping pixels into high-level semantic representation [30, 35, 9, 36] where semantic meaningful actions are represented as distributions over whole scene. Such semantic actions can be directly interpreted by humans. For instance, the semantic actions learned from a typical traffic junction are shown in Figure 1.4. We can easily name some actions as 'horizontal traffic flow', 'vertical traffic flow', 'traffic turning left', etc. The discovery of semantic representations from low-level enables projecting complex activities/interactions of multiple objects into a lower-dimension semantic space spanned by semantic actions. As a result, direct comparison between activities in the same scene at different time is possible. We show an example of this procedure in Figure 1.5.

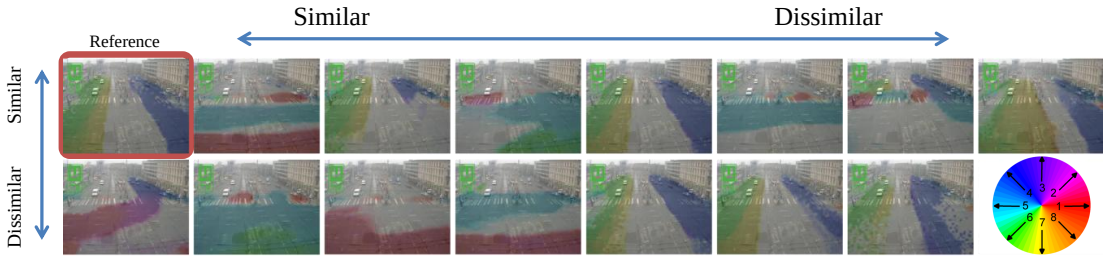


Figure 1.4: Semantic actions learned in an example scene. All actions are sorted according to the similarity to the first one in the top left. Colour overlapped on each frame indicates the direction of object movement.

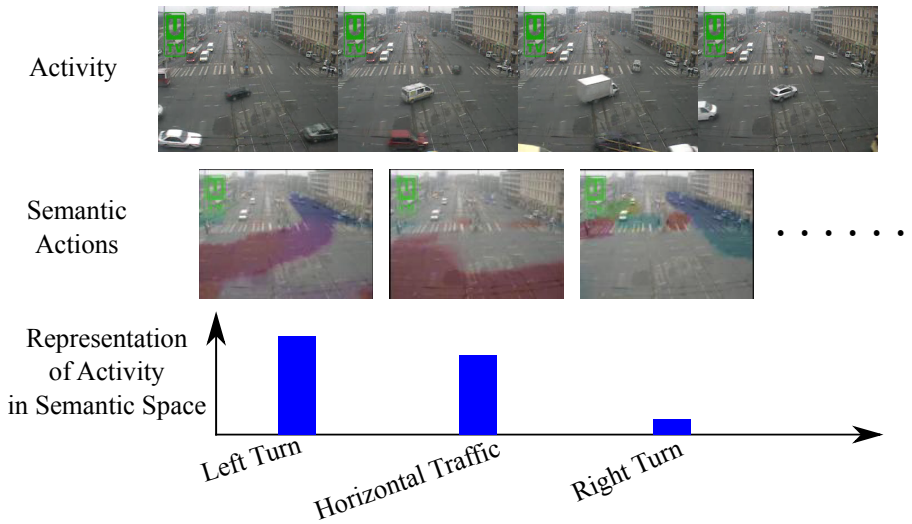


Figure 1.5: Activities/Interactions are represented as a combination of learned semantic actions.

Crowd behaviour analysis is important in video surveillance for public security and safety. It has drawn increasing attention in computer vision research over the past decade [32, 30, 37, 38,

10]. Most existing methods have been focused on developing robust and discriminative crowd scene representation [32, 30, 37]. With few exception, semantic representation has been applied to crowd behaviour analysis [38, 10]. Shao et al.[10] created a list of binary crowd/group behaviour attributes, e.g. 'outdoor', 'pedestrian', 'street', etc., for crowd modelling (examples seen in Figure 1.6). The benefit of introducing of semantic attributes is obvious in that attributes are good at characterising generic properties across scenes. Therefore attribute predictors learned from known scene can be generalised to novel ones. More importantly, the attribute set comes naturally as a semantic representation, thus can be used to describe and compare crowd behaviours in different scenes and to even facilitate zero-shot crowd behaviour prediction, e.g. the Direct Attribute Prediction (DAP) model illusrated in Figure 1.7. In the DAP model, novel behaviours $\{z\}$ can be predicted via the models learned for predicting semantic attributes $\{\alpha\}$. In this thesis we are in particular interested in recognising rare but interesting crowd behaviours, e.g. 'Violence', by exploiting crowd videos with labelled semantic attributes.

Figure 1.6: A visualisation of example crowd behaviour attributes from the WWW crowd video dataset [10].

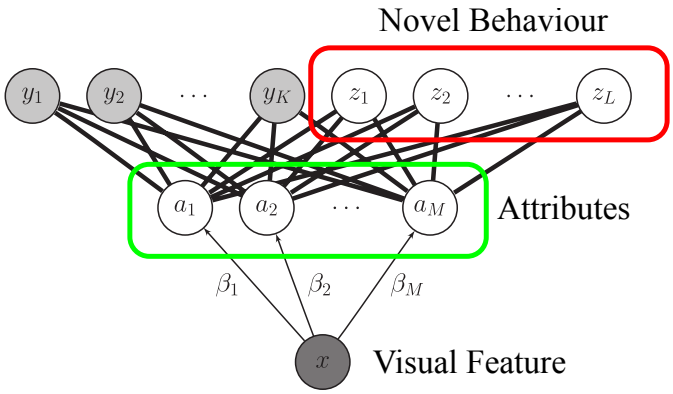


Figure 1.7: Zero-shot recognition via manually labelled semantic attributes [39].

1.3 Motivations and Challenges

In the previous sections, we briefly introduced the pipeline of exploiting semantic representation for video behaviour analysis. Nevertheless, the exploitation of semantic representation for behaviour analysis is far from perfect. Challenges exist in different aspects of action, activity and group behaviour analysis which are detailed in this section.

1.3.1 Cross/Multi-Scene Understanding and Behaviour Analysis

As introduced in Section 1.2, discovering semantic representation can facilitate the general tasks of behaviour profiling, behaviour query by example, supervised learning to annotate activities, video summarisation and anomaly detection. So far, all of these tasks have generally been addressed within a single scene (single video captured by a static camera), or a group of adjacent scenes. Compared with single scene recordings, the multi-camera surveillance network (cameras distributed over different locations) is a more realistic scenario in surveillance applications and thus of more interest to end users. An example of a typical multi-camera surveillance network is given in Figure 1.8, where surveillance videos capture mostly traffic scenes with various layouts and motion patterns. In such a multi-scene context, new surveillance tasks arise. For behaviour profiling / scene understanding, human operators would like to see which scenes within the network are semantically similar to each other, e.g. similar scene layout and motion patterns, which actions are in common – and which are unique – across a group of scenes, and how actions group into activities/interactions. Here action refers to a spatio-temporally compact motion pattern due to the motion of a single or small group of objects, e.g. vehicles making a turn, and activity/interaction refers to the interaction between multiple actions within a short temporal segment,

e.g. horizontal traffic flow with vehicles going east and west and making a turn. For query-by-example, searching for a specified example behaviour should be carried out not only within scene but also across multiple scenes. For behaviour classification, annotating training examples in every scene exhaustively is not scalable. However cross/multi-scene modelling potentially addresses this by allowing labels to be propagated from one scene to another. For summarisation, generating a summary video for multiple scenes by exploiting cross-scene redundancy can provide the user who monitors a set of cameras with an overview of all the distinctive behaviours that have occurred in a set of scenes. Multi-scene summarisation can reduce the summary length and achieve higher compression than single-scene summarisation. Combined with query-by-example (find more instances of a behaviour in a summary), a flexible exploration of scenes at multiple scales is available.

Figure 1.8: An example of multi-camera surveillance network with camera views distributed across different locations.

Despite the clear potential benefits of exploiting multi-scene surveillance, it can not be achieved with existing single-scene models [30, 9, 31, 32, 33]. These approaches learn an independent model for each scene and do not discover corresponding activities or behaviours across scenes even if they share the same semantic meaning. This makes any cross-scene reasoning about activities or behaviours impossible. In order to synergistically exploit multiple scenes in surveillance, a multi-scene model with the following capabilities is required: (1) learning an se-

mantic action representation that can be shared across scenes; (2) model activities/interactions with the shared representation so they are comparable across scenes; and (3) generalising surveillance tasks to the multi-scene case, including behaviour profiling/scene understanding, cross-scene query-by-example, cross-scene classification and multi-scene summarisation. However this is intrinsically challenging for three reasons:

1. **Semantic Representation of Behaviours**

   Constructing a semantic representation of behaviours is necessary for any further analysis. This is, however, non-trivial particularly in surveillance scenes. Firstly, it is hard to manually decide what are the basic semantic actions or how many semantic actions exist in a scene. As a result, it is highly desirable to develop an algorithm to automatically discovery semantic actions. Secondly, the semantic behaviour representation should be robust to noise in low-level visual features and able to transfer across scenes. All these requirements make the task of modelling semantic representation of behaviours challenging.

2. **Computing Action/Scene Relatedness**

   Determining the relatedness of action/scenes is critical for cross/multi-scene modelling because naive information sharing between viewpoint change can easily distort semantic similar actions [40] and insufficiently related scenes can easily result in 'negative transfer' [41]. However, the relatedness of action/scenes is hard to estimate because the appearance of elements in a scene, e.g. buildings, road surface markings, etc., is visually diverse, and strongly affected by camera view, making appearance-based similarity measurement unreliable. For comparison of events from different viewpoints, e.g. individual vehicle making left turn, any measurement should not be computed before the viewpoints effect is removed. For comparison of scenes, similarity measurement based on motion is less prone to visual noise in surveillance applications.

3. **Selective Sharing of Information**

   Large multi-camera surveillance networks covers various types of scenes. Some scenes are totally unrelated which means they convey different semantic meanings to a human. However, more subtly, even between similar scenes, there may be some activities in common and other activities that are unique to each. Learning a large universal model in this situation is prone to over-fitting due to the high model complexity. Hence a model that discovers (un)relatedness of scenes and selectively shares activities between them is necessary.

4. **Constructing a Shared Semantic Representation**

   Within related scenes, a shared action representation needs to be discovered in order to exploit their similarity for cross-scene query-by-example and multi-scene summarisation. Both common and unique actions should be preserved in this process to ensure the ability of discovering not only the commonality but also the distinctiveness between scenes.

1.3.2 Zero-Shot Action Recognition

An emerging approach to ZSL is *unsupervised* semantic embeddings [26, 42, 27, 23, 43, 28, 29]. Unsupervised semantic embedding spaces refer to intermediate representations which can be automatically constructed from existing unstructured knowledge-bases (such as wikipedia text). The most common approaches [26, 27, 28, 29] are to exploit a distributed vector representation of words, e.g. word-vector [24]. Regressors are trained on the known dataset to map low-level visual features into this semantic embedding space. Zero-shot recognition is subsequently performed by mapping novel category visual instances to the embedding space via the regression, and matching these to the vector representation of novel class names, e.g. by nearest neighbour. Several properties make the embedding space approaches favourable: (1) a manually pre-defined attribute ontology is not needed as embedding space is learned in an unsupervised manner; (2) novel categories can be defined trivially by *naming* them, without the requirement to exhaustively define each class in terms of a list of attributes – which grows non-scalably as the breadth of classes to recognise grows [27, 29]; and (3) semantic embedding allows easier exploitation of information sharing across datasets [23] because category names from multiple datasets can be easily projected into a common embedding space, while attribute spaces are usually dataset specific, with datasets having incompatible attribute schemas, e.g. UCF101 [44] and Olympic Sports [20], have disjoint attribute sets). However, the semantic embedding ZSL approach is far from perfect due to poor performance in generalising regressors from known to unknown categories and inefficient use of extra knowledge.

1. **The Domain Shift Problem for ZSL of Actions**

   Although semantic embedding based ZSL is an attractive paradigm, it has rarely previously been demonstrated in zero-shot action recognition. This is in part because of the pervasive challenge of learning mappings that generalise across the train-test semantic gap [22, 45]. In ZSL, the train-test gap is more significant than conventional supervised learning because

the training and testing classes are *disjoint*, i.e. completely different without any overlap. A serious *domain-shift* [41] problem results: mapping from low-level visual features to semantic embedding trained on a known class data will generalise poorly to novel class data, since the data distributions for the underlying categories are different. This violates the assumptions of supervised learning methods and results in poor performance. The domain shift problem – analysed empirically in Fu et al.[22] and Dinu et al.[46], and theoretically in Romera-Paredes and Torr [45] – is worse for action than still image recognition because of the greater complexity of categories in visual space-time features and the mapping of space-time features to semantic embedding space.

2. **Exploiting Extra Knowledge for Better ZSL**

   Transfer learning [41] has become a powerful tool towards many computer vision problems. A general idea of transfer learning is to utilise the data, model or any knowledge observed/learned from other domains/tasks to help the learning problem in the target domain/task. In the context of action recognition, transfer learning has already been adopted to improve performance for cross-view action recognition [47, 48], multi feature modality [49] and boosted cross-domain classification [50]. Most existing works tackled the problem of how to share information between view-points, feature modalities, categories and a pair of datasets. Although substantial improvement have been made, none of them have fully addressed the problem of how to systematically exploit the numerous datasets accumulated by the action recognition community over the last decade. We consider it is of great benefit to treat multiple datasets collectively rather than individually to assist learning problem in a more specific domain/task. This is because we can easily expand the known categories by jointly considering multiple datasets. Nevertheless, naively expanding training set with additional datasets may not be the optimal solution, as it does not take into account the (dis)similarity between the extra incorporated data and the target classes for recognition, thus risking negative transfer [41]. A good way to incorporate additional training data should be calibrated and piecewise.

3. **Multi-Variate Regression**

   As the standard procedure in ZSL, we have to learn a mapping between visual features and semantic embeddings which is a standard multi-variate regression problem. Most ZSL methods learn each dimension of this mapping *independently* – whether semantic

embedding is discrete as in the case of attributes [19, 29], or continuous as in the case of word vectors [26, 28]. This strategy is likely to overfit to the training classes because it treats each dimension of the label in semantic embedding independently despite the labels living on a non-uniform manifold [51] and many independent mappings result in a large number of parameters to be learned. We denote this conventional approach as Single-Task Learning (STL) due to the independent learning of mappings for each attribute/word-vector dimension. The STL approach is not an optimal solution to visual-to-semantic embedding due to the large number of variables in the model to fit. Therefore, a better learning strategy is needed to consider the relation between dimensions of word-vector embedding and further reduce variables to learn.

1.3.3 Zero-Shot Learning for Multi-Label Crowd Behaviour Analysis

Conventional crowd behaviour analysis depends on robust and discriminative feature design and manually annotating crowd attribute [10]. This pipeline is limited for scaling up to recognising ever increasing number of behaviour types of interest, particularly for recognising crowd behaviours of no training examples in a new environment. Firstly, conventional methods rely on exhaustively annotating examples of every crowd attribute of interest [10]. This is often implausible nor scalable due to the complexity and the cost of annotating crowd *videos* which requires spatio-temporal localisation. Secondly, many crowd attributes may all appear simultaneously in a single video instance, e.g. '*outdoor*', '*parade*', and '*fight*'. To achieve *multi-label* annotation consistently, it is significantly more challenging and costly than conventional single-label multi-class annotation. Moreover, the most interesting crowd behaviours often occur rarely, or have never occurred previously in a given scene. For example, crowd attributes such as '*mob*', '*police*', '*fight*' and '*disaster*' are rare in labelled crowd videos, both relative to others and in absolute numbers. Given that such attributes have few or no training samples, it is hard to learn a model capable of detecting and recognising them using the conventional supervised learning based crowd analysis approach.

To overcome exhaustive annotating crowd attributes and conquer the multi-label nature of crowd behaviours, we investigate and develop methods for zero-shot learning (ZSL) based crowd behaviour recognition. Although the zero-shot learning assumption and pipeline for crowd behaviour recognition is similar to that for action recognition, we find it difficult to directly adopt the pipeline for action recognition. This is due to the inherent challenges in ZSL crowd behaviour

recognition.

1. **Characterising Crowd Videos**

   Crowd videos are intrinsically difficult to model because of the complex and cluttered scenes, e.g. Figure 1.1(c). This characteristic of crowd video rules out the possibility of tracking-based models which relies on segmenting individual person/object [52]. Unsupervised scene modelling approach does not generalise to crowd analysis because the scene-specific nature prohibit the learned model to recognise crowd behaviours in new scenes. Therefore, a good visual feature is important for characterising crowd video.

2. **Multi-Label Crowd Attributes**

   Crowd scene videos are inherently multi-labelled. There are almost always multiple attributes concurrently exist in each crowd video instance. The most interesting ones are often related to other non-interesting attributes. Thus we wish to infer these interesting attributes/behaviours from the detection of non-interesting but more readily available attributes. However this has not been sufficiently studied in crowd behaviour recognition, not to mention in the context of zero-shot learning. It has been shown that in a *fully supervised setting*, exploring co-occurrence of multi-labels in a common context can improve the recognition of each individual label [53, 54, 55]. For example, the behavioural attribute '*protest*' [10] is more likely to occur in '*outdoor*' rather than '*indoor*'. Therefore, recognising the indoor/outdoor attribute in video can help to predict more accurately the '*protest*' behaviour. However, it is not only unclear how, but also non-trivial, to extend this idea to the ZSL setting. For instance, predicting a previously unseen behaviour '*violence*' in a different domain [56] would be much harder than the prediction of '*protest*'. As it is unseen, it is impossible to leverage the co-occurrence here as we have no *a priori* annotated data to learn their co-occurring context.

1.4 Approaches

To tackle the challenges summarised as before, we proposed semantic space based approaches for video behaviour analysis. Specifically, we proposed the following solutions.

1.4.1 Cross/Multi-Scene Understanding with Semantic Action Discovery and Matching

For static surveillance scenes, as discussed in Section 1.3.1, it is important to discover a semantic representation of behaviours, to define an appropriate measure between semantic events or scenes, to selectively share or transfer knowledge between scenes and to build a shared representation across multiple scenes. In this section, we detail the specific approaches proposed to answer the challenges.

Unsupervised Semantic Action Modelling

We propose to learn semantic representations of behaviours, referred to as semantic actions, in an unsupervised manner to overcome the difficulties explained in Section 1.3.1. Crucially, we believe modelling semantic actions as probabilistic models is robust to low-level visual noise and preserves the geometry of actions which is essential for cross-scene matching. Specifically, for clean and birds-eye view scenes, as studied in Chapter 3, we do multi-object tracking to extract individual motion information and then learn statistical models, e.g. Gaussian Mixture Model (GMM), to represent typical semantic actions. For busy and heavily occluded views, as studied in Chapter 4, we compute optical flow to extract basic motion and apply hierarchical bayes networks, e.g. Latent Dirichlet Allocation (LDA), to model the semantic actions.

Geometrical Alignment for Computing Semantic Action/Scene Relatedness

To share/transfer knowledge across scenes captured by different cameras, the viewpoint change is the first obstacle. Since any measurement of events and scenes should be invariant to viewpoint, we propose to employ a geometric alignment to mitigate the viewpoint change. For comparison between semantic actions modelled as probabilistic models, e.g. GMM and LDA, we employ the Kullback-Leibler Divergence (KLD) measure. Whilst the viewpoint change still prohibits the semantic measure. To overcome this challenge, we optimise KLD over similarity transforms in Chapter 3, thus achieving a similarity-transform invariant distance metric.

With the geometric alignment, semantic similar actions can be alignment and measured with KLD. Nevertheless, the per-action geometrical alignment is overly localised and ignored the layout of a scenes and interactions between actions. It is thus a too strong assumption for aligning two scenes. Moreover, considering the fact that most surveillance cameras are installed upright and there are many classic types of traffic scenes, we consider aligning two surveillance scenes with a much relaxed assumption in Chapter 4, by scaling and translation only.

Shared Semantic Space Discovery with Multi-Stage Clustering

As discussed, multi-scene behaviour analysis requires constructing a shared representation across scenes. It is also vital to selectively share information during this process, i.e. learning model within semantic related scenes. Specifically, in Chapter 4 we first represent each scene with a low-dimensional semantic representation, through learning a unsupervised probabilistic model. Using a topic-based representation allows us to reduce the impact of pixel-noise in discovering activity and scene similarity. We next group *semantically* related scenes into a scene cluster by exploiting the correspondence of actions between different scenes. Finally, scenes within each cluster are projected to a shared representational space by computing a *shared action topic basis* (STB), shared among all scenes but also allowing each scene to have unique topics if supported by the data. Behaviours in each scene are represented with the learned STB.

1.4.2 Transductive Zero-Shot Action Recognition with Prioritised Data Augmentation

As discussed in Section 1.3.2, domain shift problem has limited the performance of zero-shot action recognition. It is also unclear how to best exploit the available related dataset to help zero-shot action recognition. To resolve these issues, we formulate a transductive zero-shot learning (ZSL) strategy to exploit the distribution of unlabelled testing data with prioritised data augmentation to selectively expand training dataset.

Transductive ZSL

In Chapter 5, we explore three transductive solutions to ameliorate the domain shift challenge in ZSL for action recognition. Here we refer 'transductive' to the assumption that we have the access to unlabelled testing data. The first strategy we consider aims to improve the generalisation of the embedding space mapping. We explore **manifold regularization** (aka semi-supervised learning) to learn a regressor which exploits a regularizer based on the testing/unlabelled data to learn a smoother regressor that better generalises to novel testing classes. Manifold regularization [57] is established in semi-supervised learning to improve generalisation of predictions on testing data, but this is more important in ZSL since the gap between training and testing data is even bigger due to disjoint categories. We also evaluate two existing post-processing heuristics to reduce the effect of domain-shift in ZSL. These include **self-training**, which adapts test-class embedding representations based on unlabelled testing data to bridge the domain shift [22] and **Hubness correction** which re-ranks the test-data's match to novel class descriptions in order to

avoid the bias toward 'hub' categories induced by domain shift [46].

Exploiting Additional Data

From a data-, rather than model-centric perspective, studies have attempted to improve the generalisation of ZSL methods by augmenting the auxiliary dataset with additional datasets containing a wider array of classes and instances [23]. In this thesis, data augmentation means exploiting additional data in a wider context from multiple data sources, in contrast to synthesising more artificial variations of one dataset as in deep learning. The idea is that including a broader additional set should provide better coverage of the visual feature and label embedding spaces, therefore helping to learn a visual-semantic mapping that better generalises to target classes, and thus improves performance when representing and recognising target classes. However, existing studies on exploring this idea have been rather crude, e.g. simply expanding the training dataset by blindly concatenating auxiliary set with additional data. This is not only inefficient but also dangerous, because it does not take into account the (dis)similarity between the extra incorporated data and the target classes for recognition, thus risking *negative transfer* [41]. In this thesis, we address the issue that auxiliary and target data/categories will have different marginal distributions. We selectively re-weight those relevant instances/classes from the auxiliary data that are expected to improve the the visual-semantic mapping in the context of the specific target instances/classes to be recognised. We formulate this prioritised data augmentation as a domain adaptation problem by minimizing the discrepancy between the marginal distributions of the auxiliary and target instances and/or classes. To achieve this, in Chapter 6, we propose an importance weighting strategy to re-weight each auxiliary instance in order to minimise the discrepancy. Specifically we generalise the classic *Kullback-Leibler Importance Estimation Procedure* (KLIEP) [58, 59] to the zero-shot learning problem.

Multi-Task Semantic Embedding

It has been discussed in Section 1.3.2 that modelling the visual-to-semantic mapping for each individual dimension is not an optimal solution due to the potential correlation between dimensions. To solve the multi-variate regression problem in a more effective way, we propose to learn the mapping in a multi-task learning (MTL) manner in Chapter 6. We assume the regressor for each individual dimension of semantic embedding can be represented as a combination of fewer shared regressors. With this assumption, the models for each dimension are forced to correlate with each other and we have overall fewer parameters to fit than modelling separately. The re-

sulting visual-to-semantic mapping is more robust to the domain shift between ZSL training and testing classes. As a helpful by-product, the MTL mapping, provides a lower dimensional latent space in which the nearest neighbour (NN) matching required by ZSL can be better performed [60] compared to the usual higher dimensional label semantic embedding space.

1.4.3 Context-Aware Zero-Shot Learning for Crowd Behaviour Recognition

As discussed in Section 1.3.3, we overcome the difficulties in zero-shot crowd behaviour recognition by introducing a context-aware zero-shot learning model in Chapter 7. This model depends on the statistics of multi-label co-occurrence. Specifically, we firstly calculate the co-occurrence between every pair of known labels and encode this co-occurrence via a bilinear mapping with the help of label name word-vector. Then we train recognisers for each known label on the training set. In the testing phase, we generalise the learned bilinear mapping to predict the relation between known and unknown labels. With this label relation we can predict the confidence for each unknown label through the predictions of known recognisers. We owe the improved performance to encoding the co-occurrence between known semantic attributes and the exploitation of word-vector representation of attribute names.

1.5 Contributions

The contributions of this thesis are mainly towards exploiting different semantic spaces, e.g. unsupervised text-based word-vector, unsupervised semantic action and manually annotated crowd attributes, for different video behaviour analysis tasks and how to solve the subsequent issues brought by cross/multi-scene analysis, zero-shot learning and multi-label learning. In specific, we make the following contributions:

1. We introduce a novel and challenging problems of joint multi-scene modelling and analysis. To solve these problems, a framework is proposed by discovering similarity between semantic actions and scenes, clustering scenes based on semantic similarity and learning a shared representation within scene clusters. All of these are based on discovering semantic action representations from repetitive visual data. We show how to exploit this novel structured multi-scene model for practical yet challenging tasks of cross-scene query-by-example and behaviour annotation. We further exploit this model to achieve multi-scene video summarisation, achieving compression beyond standard single-scene approaches.

Finally, we introduce a large multi-scene surveillance dataset containing 27 distinct views from distributed locations to encourage further investigation into realistic multi-scene visual surveillance applications.

2. For zero-shot action recognition, we propose to exploit unsupervised text-based word-vector as the semantic embedding space to bridge the gap between known and unknown categories. we further explore jointly four mechanisms for expanding ZSL by addressing its domain-shift challenge, including three transductive learning strategies - manifold regularization, self-training and hubness correction and data augmentation. We also provide new insight, for the first time, into the underlying factors affecting the efficacy of ZSL.

3. To improve upon the conventional single-task learning methodology, we advocate a Multi-Task Learning (MTL) [61, 62, 41] regression approach to mapping visual features and their semantic embeddings. By constraining the mapping parameters of each learning task to lie closely on a low-dimensional manifold, we gain two advantages: (1) exploiting the relation between the response variables (dimensions of the label embedding); and (2) reducing the total number of parameters to fit. The resulting visual-to-semantic mapping is more robust to the domain shift between ZSL training and testing classes. As a helpful byproduct, the MTL mapping, provides a lower dimensional latent space in which the nearest neighbour (NN) matching required by ZSL can be better performed [60] compared to the usual higher dimensional label semantic embedding space.

4. Inspired by the insight revealed in zero-shot action recognition, we note that naive data augmentation does not address the potential mismatch between auxiliary and source dataset. A brute-force data aggregation could deteriorate the performance on target testing dataset. In this thesis, we tackle this issue by selectively re-weighting those relevant instances/classes from the auxiliary data that are expected to improve the the visual-semantic mapping in the context of the specific target classes to be recognised.

5. In zero-shot crowd behaviour analysis, we, for the first time, investigate zero-shot learning for crowd behaviour recognition to overcome the costly and semantically ambiguous multi-label video annotation. Moreover, we propose a contextual learning strategy which enhances novel attribute recognition through context prediction by estimating attribute-context co-occurrence with a bilinear model. Finally, a proof-of-concept case study is pre-

sented to demonstrate the viability of transferring zero-shot recognition of violent event cross-domain with very promising performance

1.6 Outline

This thesis consists of 8 chapters: **Chapter 2** reviews the general pipeline for video behaviour analysis, recent development in semantic space discovery and related machine learning strategies involved. **Chapter 3** introduces a pipeline to deal with cross-scene semantic behaviour recognition. In specific, the semantic action model learned from one scene can be re-used to interpret a new scene without re-training and finally facilitate abnormal behaviour detection. **Chapter 4** studies a multi-scene behaviour analysis framework. A multi-layer clustering algorithm is proposed to group semantic related scenes and discover semantic action representations shared by related scenes. Multi-scene surveillance tasks including multi-scene profiling, cross-scene classification, cross-scene query and multi-scene summarisation are enabled by our framework. **Chapter 5** tackles the domain-shift problem in zero-shot learning by transductively exploiting the unlabelled testing data and re-using additional related labelled dataset (data augmentation). The integration of transductive approaches and data augmentation is able to improve upon conventional zero-shot learning performance on human action recognition. Importantly, the insight we make in this work inspires a more sophisticated exploitation of additional data. **Chapter 6** proposes two methods to further address the domain-shift and data augmentation. The multi-task learning approach is proposed to improve upon conventional single-task regression approach. Moreover, we also introduce an auxiliary data re-weighting scheme to get rid of 'negative transfer' in data augmentation. **Chapter 7** considers a possibility of applying zero-shot learning to crowd behaviour analysis. In particular, a multi-label zero-shot learning scheme is proposed in response to the multi-label nature of crowd attribute prediction. **Chapter 8** summarises this thesis and points out the directions for future work. Finally, we provide a clear summary of each chapter in Table 1.2.

Table 1.2: A brief summary of each chapter

	Visual Data	Semantic Spaces	Challenges
Chapter 3 Cross-Scene Semantic Behaviour Recognition	Static Surveillance	Visual Induced	Geometric Alignment; Selective Transfer
Chapter 4 Semantic Space Discovery for Multi-Scene Behaviour Analysis	Static Surveillance	Visual Induced	Scene-Level Relatedness; Selective Sharing; Shared Representation
Chapter 5 Semantic Space for Zero-Shot Action Recognition	Human Action & Event	Text Induced	Domain Shift; Exploiting Extra Data
Chapter 6 Multi-Task Semantic Embedding with Prioritised Data Augmentation	Human Action	Text Induced	Multi-Variate Regression; Exploiting Extra Data
Chapter 7 Zero-Shot Crowd Behaviour Analysis	Crowd Videos	Manually Labelled Attribute	Multi-Label Zero-Shot Learning

Chapter 2

Literature Review

Video behaviour analysis consists of multiple components from low-level feature processing to mid-level semantic representation learning as well as high-level behaviour analysis. In this chapter, we briefly review the existing works relevant to our problems and techniques by examining each individual components:

1. Visual Feature Representation. This include multiple basic computer vision techniques including optical flow, tracking, visual descriptors and feature encoding.

2. Semantic Representation. The semantic representation induced from a collection of low-level visual features, external text corpus and manually annotated attributes for video behaviour analysis.

3. Video Behaviour Analysis. The basic tasks for behaviour analysis include surveillance scene understanding, action recognition and crowd behaviour analysis.

4. Learning Strategies. Different learning techniques involved in thesis.

Extensive reviews on general human activity analysis has been made in Aggarwal and Ryoo [3] and Poppe [63] covering visual feature discussion and different approaches to activity recognition. A study with focus on surveillance has been conducted by Hu et al.[64]. In particular, crowd behaviour analysis in surveillance context has been addressed by Gong et al.[65] and Li et al.[66]. For reviews on general machine learning techniques, Pan et al.[41] made a thorough

discussion on transfer learning with a case study, whilst a comparison study was conducted by Zhang et al.[53] covering all aspects of multi-label learning.

2.1 Visual Feature representation

Construct a reliable and discriminative feature is critical to the success of any high-level semantic tasks. In contrast to image-based features, the appearance and motion based features can provide complementary information for video content. Therefore both motion feature, e.g. trajectory and optical flow, and appearance features, e.g. HOG and SIFT, have been successfully applied in video analysis. Beyond the existing hand-crafted visual features, deep neural networks [67, 68, 69] have shown great potential in video behaviour recognition with both human action recognition and complex event detection. In the following section, we review widely used low-level feature extraction and feature encoding approaches for video behaviour analysis.

2.1.1 Low-Level Features

The low-level features are usually extracted directly from video sequence and serve as the most primitive information of video content. We divide the low-level features into 3 groups, object-based motion feature, pixel-based motion feature and appearance-based motion feature. The former two in general capture the dynamics of video content while ignoring the appearance of video content. The latter one is usually extended from image-based features and is able to capture the appearance and motion of video content but with more computation cost. In modern computer vision frameworks, different low-level features are often used in conjunction to provide more discriminative features.

Object-based Motion Features

The object-based motion feature assumes good segmentation of object of interest from the scene which are fed into latter analysis. Trajectory with corresponding bounding-box induced by multi-object tracking are the most widely adopted object-based feature [70, 71, 72, 31]. Tracking objects usually involves constructing object representation, object detection, object tracking/association and possibly handling occlusion [73]. For object representation, various object shape models including points [74], primitive geometric shapes, e.g. rectangle, [31] are widely adopted assumptions. Whilst, the object feature is usually selected as colour and texture [73]. Object detection aims to provide candidate objects proposals for tracking/association. Detecting

interesting points [75, 76] and more recently densely sampling points [15] have been applied to tracking. Nevertheless, the interesting or dense points hardly correspond to meaningful individual object, e.g. single person or vehicle. For better object segmentation, background subtraction [77, 78] provides an alternative way to segment foreground objects. Though being able to provide object detections, the detected foreground itself is not a motion feature without tracking. Provided with more computation capability, direct object detection is viable for tracking face [79] and pedestrian [80]. With object detections, tracking is implemented with dynamic model, e.g. Kalman Filter [81], and data association. The resulting position, velocity and the size of bounding box [31] serve as the basic representation for behaviour analysis. These representations can be further fed into learning higher level models by clustering [82, 31] or statistical models [36].

However, the real-world surveillance scenes are always cluttered, crowded and with occlusion. All these attributes prohibit accurate foreground segmentation and correct data association. To cope with the challenges in real scenes, people have proposed alternative approaches to extract motion feature than tracking-based one. We introduce the pixel-based motion feature in the following section.

Pixel-based Motion Features

Pixel-based features are in contrast to object-based ones in not requiring accurately segmenting foreground object. This overcomes the challenges brought by cluttered and crowded surveillance scenes. In specific, we discuss two prevailing pixel-based motion feature here, background subtraction and optical flow.

As discussed in the previous section, background subtraction provides object detection for tracking. However, tracking is not reliable when foreground objects are frequently occluded which is difficult for data association and/or highly crowded where segmentation is challenging. An alternative way to use background subtraction is by tracking-free [8]. This approach detect foreground pixels by modelling the background [77]. Detected foreground pixels are grouped into local blobs. Descriptions of the blobs' position, geometry and velocity collectively form motion features. This foreground pixel-based features do not require object segmentation and tracking thus is able to deal with more complex and crowded scenes. However, the grouped blobs does not necessarily correspond to individual objects.

Apart from background subtraction, optical flow is an alternative pixel-based feature [30, 9, 36]. Optical flow captures the motion of pixels between consecutive frames [83]. Scenes are

usually segmented into cells [9] and optical flow is computed and averaged at the cell level to avoid outliers and save computation cost. Pixels with flow magnitude higher than a threshold are considered as motion area. Optical flow can be modelled in conjunction with background subtraction to incorporate stationary pixels [9]. Similar to background subtraction, optical flow is able to cope with extremely crowded and cluttered scenes while being more demanding in computation. Due to the ability to estimate the motion of foreground, optical flow is usually complementary to background modelling and used together for better feature representation [9, 8]. Nevertheless, computing reliable optical flow requires stable video stream with relatively high frame-rate. A low frame-rate video with large displacement of object between frames could cause low accuracy in optical flow [84].

Both the object-based and pixel-based features are based purely on the motion clues of objects while ignoring the visual appearance. Therefore, these features are less discriminative in more complex behaviours where texture and visual appearance dominate. More importantly, the original form of tracking, background modelling and optical flow are designed for static scenes. Hence, it is not straightforward, e.g. tracking, optical flow, or less likely, e.g. background modelling, to extract meaningful features from non-statics scene. To tackle these issues, people have proposed appearance based motion features developed from traditional image descriptors working together with pixel-based motion features.

Appearance-based Motion Features

To cope with more complex behaviours and non-static scenes appearance-based motion features have been derived from traditional static image descriptors [13, 85, 86, 87]. Inspired by image interest point detection, Laptev [13] generalised interest point detection to extracting video features. Harris interest points are first detected [88] in generalised space-temporal space. Both appearance-based Histograms of Oriented Gradient (HOG) and motion-based Histogram of Optical Flow (HOF) are then computed on the detected 3D video patches and concatetnated to form visual features. By treating the video as 3D volumetric data, SIFT descriptor [76] was extended to extracting descriptors from videos [85]. The spatio-temporal nature can be simultaneously captured by 3D SIFT. As an extension of HOG descriptor [80], HOG3D [86] was proposed to compute gradient in 3D volume and quantize gradient vectors into histogram representation. The pure dependency on spatio-temporal 3D gradients avoids the expensive computation of optical flow which is used on STIP with HOF descriptor. More recently, dense trajectory [89, 15] has

been proposed for human action recognition particularly in non-static scenes with state-of-the-art performance. In contrast to interest point detection, dense trajectory feature densely sample points from each frame and track dense points across frames with median filter. Both appearance and motion based descriptor, HOG, HOF and Motion Boundary Histogram (MBH), are computed along the trajectories and concatenated into raw descriptors. The advantages of densely tracking points are: (1) tracking points based on optical flow avoids the challenge of multi-object data association; (2) densely sampling points ensure good coverage of the scene; and (3) MBH descriptor and further homography estimation [90] are able to mitigate the camera motion thus improving performance on non-static scenes.

In summary, given static scenes (mostly in surveillance videos) object-based feature (tracking) is preferred with sparse and less occluded views [70, 71, 72, 31]. This will in particular benefit anomaly detection where an individual object, e.g. vehicle, is wanted to be identified. With crowd, cluttered and occluded static views, pixel-based features (background modelling and optical flow) are preferred for their robustness to occlusion [30, 9, 36]. However the individual object is hard to segment with these features. When analysing behaviours in non-static scenes and/or scene-independent tasks(mostly in human action or complex video event) appearance-based features are preferred due to the capability to capture both motion and appearance information and being robust to camera motion [89, 15]. Nevertheless, appearance-based features are also suitable for processing crowd and cluttered views in surveillance-like videos [10]. In this thesis, we employ multi-object tracking feature for bird-eye view surveillance scenes in Chapter 3, optical flow feature for crowd and partially occluded surveillance scenes in Chapter 4 and dense trajectory feature for both human action and surveillance-like crowd views in Chapter 5, 6 and 7.

2.1.2 Feature Encoding

The purpose of feature encoding is to generate consistent and comparable vectorized representations of motion features before being fed into learning algorithms.

Trajectory Preprocessing

Due to the inherent time-varying nature, trajectories are always of different length. Therefore it is necessary to normalise or project trajectories into another space. Normalisation manipulate original trajectories and produce processed trajectories with equal length. Zero-padding [31] adds

extra zero points to the original trajectory to compensate the gap. Track-extension [64] applied simulated tracking using the dynamics at the end of trajectory until the normalised length. These two strategies are rather simple; however, the processed trajectory loses some dynamics. As an alternative way, re-sampling combined with smoothing [91, 64] preserve the dynamics of trajectory without too much computation.

Bag of Words

It is essential to encode low-level visual descriptors into fixed-length and normalised feature which are ready to be used for learning algorithms. Prevailing feature encoding approaches including Bag of Words (BOW), Fisher Vector (FV) [92] and VLAD [93]. In general, BOW is the most basic but computation efficient approach. It firstly generates a fixed codebook by either learning from data or manually defining a codebook. For non-static scenes, raw descriptors are usually scene independent. Thus Kmeans clustering is commonly employed to discover a representative set of raw descriptors as codebook. When static scenes are considered, a grid is pre-defined to characterise basic motions descriptors, e.g. optical flow, and each cell with quantized directions form a codebook [94, 9]. Given learned or pre-defined codebook, low-level features are assigned with one of the integer index according to the distance to each of the codebook entry. A histogram over codebook is constructed to represent the feature of a long document or video as our problem. A Fisher Vector is further improved over the simple BOW strategy by taking the first derivative information into account. To encode feature with a fisher vector, Gaussian Mixture Model (GMM) is first learned from training data in contrast to Kmeans clustering. The derivative of joint probability of low-level descriptor w.r.t. GMM means and covariances are computed and additively accumulated as a fisher vector encoded feature. FV often presents superior performance to BoW due to the consideration of first order information (derivative) and the relation of one descriptor to all gaussians rather than assigning to a single gaussian as with Kmeans. It is also worth noting that FV generates much higher dimension encoded feature than BOW which can be resolved by the kernel trick [95].

2.1.3 Deep Feature

Deep learning has been proved to produce the state-of-the-art performance in still image recognition on Image-Net challenge [96, 97]. Deep learning can be loosely defined as a class of machine learning techniques that exploit many layers of non-linear data transformation for supervised or

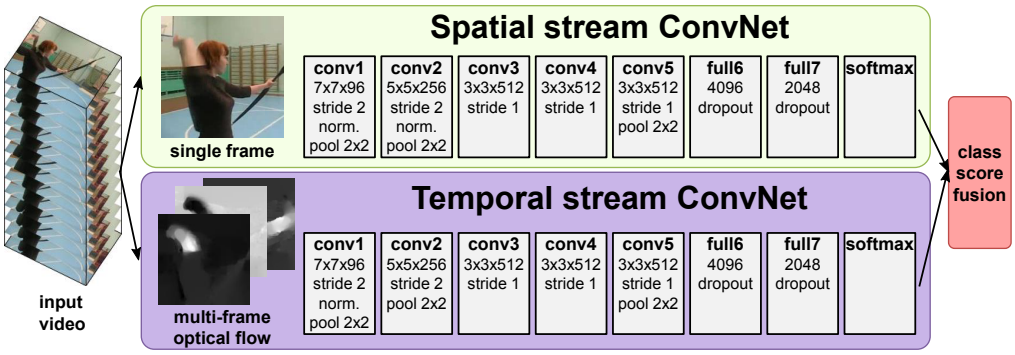


Figure 2.1: The two-stream CNN network with separate appearance and motion networks [68].

unsupervised tasks [98]. As the non-linear transformation is realised by artificial neural network, the architecture is usually referred to as deep neural network. The advantage of deep neural network is largely attributed to the ability to model highly non-linear transformation. As many of neurons are connected, deep neural network has way more parameters to learn than traditional machine learning models. For instance, AlexNet [96] is designed with 60 millions parameters while the traditional bag of words encoding SVM model could only have 1 millions parameters in total [99]. Because of the superior performance of deep learning, people have recently started applying deep learning to improve video recognition [10, 68, 69]. In this section we briefly review the recent development of deep video feature.

Inspired by the success of convolutional neural network (CNN) in recognising still images [100], Karpathy et al.[67] initially extended CNN to video content understanding. Crucially, this attempt incorporates the CNN with not only appearance information but also complex temporal evolution which carries lots of information in video content. However the performance is not very competitive compared with hand-crafted features, e.g. dense trajectory [101], because the motion is not explicitly provided for the network. Based on this idea, a two-stream CNN was proposed by Simonyan et al.[68], as illustrated in Figure 2.1. In this framework, a separate temporal stream CNN was incorporated apart from the spatial stream to help exploit motion information. Two streams of network can be pre-trained and predict separately. The final prediction is a score level fusion of two networks. The explicit feed-in of motion as optical flow significantly improved the performance on action recognition and became the prevailing idea in deep video understanding.

In processing video content, CNN often takes a stack of video frames as input as with Karpathy et al.[67] and Simonyan et al.[68]. To generate fixed length of representations for videos, the descriptors of individual frames (often the output of certain layers of neural network) are

averaged [69]. Obviously, a lot of information is lost during the averaging process making the representation less meaningful. This has degenerated the performance of CNN on video recognition, in particular the TRECVID dataset [2] where videos are often very long. Inspired by the prevailing encoding method in hand-crafted features, Xu et al.[69] proposed to encode CNN descriptors by Fisher Vector and VLAD. The encoding approach was proved to greatly improve the performance of deep model on multimedia event detection tasks (MED) and even outperform best hand-crafted feature (dense trajectory).

Applying deep model to crowd video analysis has been studied as well [10]. Inherited from Karpathy et al.[67], both appearance and motion branches are exploited in Shao et al.[10]. To account for the inherent nature crowd video, a feature map specially designed on collectiveness, stability and conflict [38].

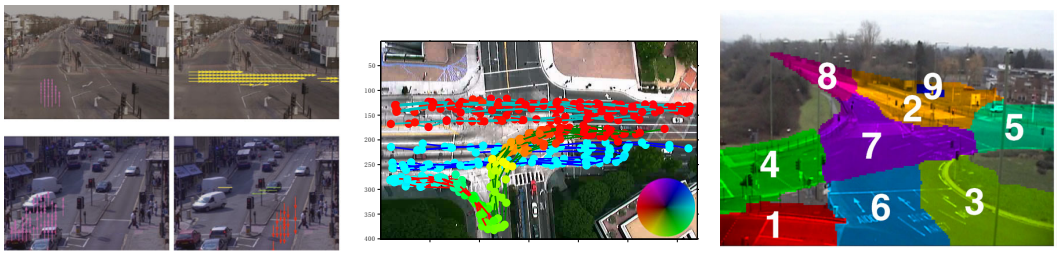
Overall, deep learning is becoming the mainstream in video content analysis due to its ability to model highly non-linear transformation and learning from large database. However, the ability of deep model depends on the size of training dataset. Therefore, on small training set like action recognition dataset, hand-crafted feature is still a good choice compared with deep model without pre-training on huge external dataset.

2.2 Discovering Semantic Representation

The semantic space can greatly benefit video behaviour analysis. Complex behaviours are usually represented upon shared semantic space to facilitate semantic analysis and information sharing between different scenes and semantic categories. Thus the semantic space works as the basic building blocks for many higher-level video behaviour analysis. In this section, we discuss three alternative ways to discover such a semantic space, namely visual induced semantics, manually annotated semantics and text induced semantics.

2.2.1 Visual Induced Semantics

The low-level visual features, e.g. object-based and pixel-based motion feature, are usually semantically meaningless, e.g. foreground pixels, or not robust, e.g. a single trajectory. Therefore it is desirable to group these visual features to higher-level semantic representation upon which more complex behaviours can be represented. A typical visual induced semantics include traffic and pedestrian flows and semantic regions as shown in Figure 2.2. Importantly, in the absence



(a) Markov Topic Model [9] (b) Gaussian Mixture Model [36] (c) Semantic Regions [8]

Figure 2.2: Examples of visual induced semantic representations of behaviours.

of manual annotations, semantic representation can be learned from visual data in an unsupervised way [30, 9, 36, 8, 102]. The procedure of semantic grouping is often framed as learning statistical models, e.g. Gaussian Mixture Model (GMM), Hidden Markov Model (HMM) and Topic Model. The learned semantic behaviours are frequently referred to as motion pattern [36] or topics [30, 9] in the literature.

Clustering from trajectory

Trajectory cluster is able to group observed trajectories into clusters and then statistical motion patterns are learned within trajectory clusters [31, 103, 104, 105]. For modelling motion patterns from trajectories, Hu et al.[31] employed two layer clustering with spatial and temporal cues respectively. Within each subset of trajectories, a series of gaussian distributions are fitted to each state of the trajectory. The number of clusters is determined by Tighness and separation criterion [106]. When trajectories are considered piecewisely, Lee et al.[104] proposed to cluster sub-trajectories to discover sub-trajectory clusters. Overall, clustering trajectories supports further modelling of statistical model for behaviour analysis.

Mixture Model

Gaussian Mixture Model (GMM) is a powerful probabilistic graphical model for modelling motion pattern due to its ability to modelling arbitrary distributions and low computation cost. GMM has been proposed to represent motion pattern by Saleemi et al.[36] and Khokhar et al.[40] where a motion pattern is represented by the combination of several weighted Gaussian distributions. The GMM is learned from raw optical flow results by Kmeans clustering and Maximum Likelihood Estimation. The benefit of this method is that it not only provide a positional information of a motion pattern, e.g. vehicles turning left at the junction, but also give probability of what is going to happen next (usually given as speed and direction) conditioned on current position.

Bayes Network

Bayes network (BN) has been widely utilised for modelling statistical motion patterns/models. As a directed graphic model, BN is able to capture the dependencies between states, e.g. the location of vehicles, of motion procedures. As a dynamic BN, Hidden Markov Model (HMM) has been widely used to model temporal evolution of state [107]. Augmented Hidden Markov Model(AHMM) have been applied to model probabilistic spatio-temporal regularities for providing visual expectations and selective attention [108]. HMM was also adopted [70] to handle time normalisation of motion event.

Probabilistic Topic Model

Probabilistic topic model (PTM) is a more recent approach towards modelling local activities [30, 9, 32]. PTM originated in natural language processing with lots of variants, e.g. Probabilistic Latent Semantic Index (pLSI) [109], Latent Dirichlet Allocation (LDA) [110] and Hierarchical Dirichlet Process (HDP) [111]. Topics as distributions over individual words are discovered by PTM from a collection of documents. Frequently co-ocurring words are more likely to be grouped into the same topic. PTM was first introduced to modelling activities in videos by treating quantized motion feature, e.g. encoded optical flow and foreground pixels/cells, as 'words' and short video clips as 'documents' [30]. The discovered 'topics' group frequently co-occurring cells and correspond to semantic meaningful actions, e.g. horizontal/vertical traffic flow [30]. With the discovered topics, video clips are further represented as profiles (fixed length vectors) on discovered topics. With the profile on same set of topics, video clips from the same scene at different time are comparable and can be used for further analysis. Within this framework, Wang et al.[30] adopted LDA and HDP to model both traffic junction and indoor crowds and achieved video segmentation, anomaly detection and semantic query. To further model the temporal relation between behaviours, Hospedales et al.[9] extended conventional LDA with a markov chain. The ordering of behaviour are thus encoded in this model.

In general, PTM models depending on pixel-based motion features are in particular robust to crowd and occluded scenes where tracking is not reliable. However, all of the existing studies operate within a specific scene. Models learned from one scene can not be re-used on or benefit new scenes due to the unique topics. Because of the importance of multi-scene understanding, this thesis proposes a novel multi-scene understanding approach and experiments on multi-scene urban surveillance network in Chapter 4.

(a) Animal attributes [39] (b) Human action attributes [20]

Figure 2.3: Examples of attributes for animal (AWA) with attributes dataset and human action dataset.

2.2.2 Manually Annotated Semantics

Instead of learning semantic representations from visual data, manually defining a list of semantics for visual data is an alternative way to construct semantic representation of visual data. Attribute, in particular, has been the dominating way to this goal.

Attribute

Attributes are manually annotated binary semantic labels. As illustrated in Figure 2.3, they usually define visual distinctive features in image [39] or atomic actions in videos [20] which are both semantic meaningful and can transcend beyond high-level categories. Manually labelled attributes have been proved to assist visual recognition tasks [20], in particular, transferring knowledge across categories (zero-shot learning) [39]. Defined as binary labels, representing visual content based on attributes is often realised as learning per-attribute classifier. Specifically, SVM classifiers have been trained for each attributes on training data [39, 20], visual contents are then represented as the probability prediction of classifiers. However, attributes are manually annotated and thus suffer from: (1) the difficulty of determining an appropriate ontology of attributes; (2) prohibitive annotation cost, in particular for videos due to their spatio-temporal nature [20]; and (3) labelling each video with a large vocabulary of attributes is particularly costly and ambiguous in contrast to annotating images [39]. All of these issues have inspired people to discover more readily available semantic representations by learning from textual data.

2.2.3 Text Induced Semantics

Apart from manually labelled attributes, unsupervised semantics have been applied to video behaviour analysis [26, 22, 23]. In particular, text induced semantics becomes the prevailing method due to the extensive research in natural language processing (NLP) and zero-cost in transferring text induced semantics to visual analysis. In text induced semantics, vector representation of words and phrases are usually learned from text corpora e.g. wikipedia document [24, 25] and video metadata [23]. We owe the benefit of learning from text to the semantic meaningful representation of each individual word or phrase. In contrast to the visual induced semantics, the textual semantics can work in conjunction with manually annotations e.g. category names, video descriptions, to facilitate learning problems beyond standard supervised learning.

Word-Vector

Word-vector [24] among many NLP models was proposed to learn vectorized representations of individual words or phrases. This model builds a vector for each tokenized word/phrase from large text corpus. A skip-gram model is adopted to predict the surrounding words given current word. The by-product of this optimization is the vector representations for all tokenized words. By exploiting the co-occurrence of words within a context, the Global Vectors for Words Representation (GloVe) [25] was proposed to learn vector representations of words. The GloVe model defines a weighted square loss between the co-occurrence of words and exponentialized inner product between word vectors. The vectors can be learned via gradient-based methods.

While, individual word can be represented by a semantic meaningful vector, it is revealed in Mikolov et al.[24] that vectors for compound names/phrases can be achieved by summing up word-vectors of individual words. Good performance in reasoning analogies e.g. 'Germany':'Berlin'::'France':'Paris' (see Figure 2.4) suggest the efficacy of additive constructing phrase vectors. However, sentence consisting of multiple words are hardly meaningful when summing up each individual words [112]. Describing visual content with a sentence of fragmented words are commonly referred to as metadata [2]. In order to construct semantic representation for such more complex textual descriptions, people have considered alternative models based on bag-of-words representation of sentences and learn topic models to reduce data dimension [23] and more intuitively training a sentence vector model [112].

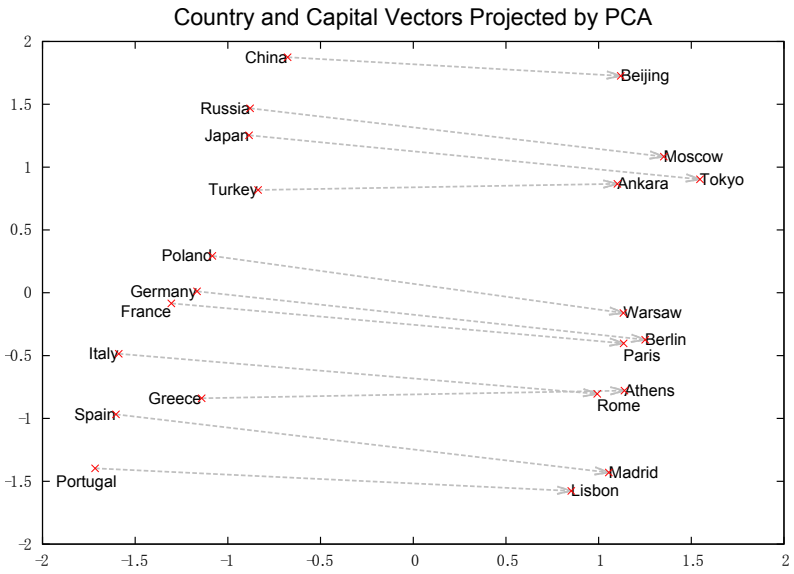


Figure 2.4: Two dimensional projection of word-vectors. This figure illustrates the ability to reason the concept of 'capital' after training on large text corpus [24].

Topic Model

Topic model [110, 113, 114] has been studied for learning vectorized representations for documents or sentences. There are two genres of topic models in terms of mathematical formation including probabilistic topic model, as introduced in the previous section, and matrix factorisation. Both topic models aim to group frequently co-occurring words into topics and then represent a document as a profile or distribution on topics. In particular, given word-document matrix the matrix factorisation models factorise into the product of two low-rank matrices, word-topic and topic-document. The conventional Latent Semantic Analysis (LSA) [113] solves this low-rank factorisation with singular value decomposition (SVD). The first unitary matrix is the word-topic representation and the last unitary matrix is the approximation of document in terms of topics. To enable modelling large scale dataset, Wang et al.[114] framed the factorisation as a regularised low-rank decomposition with L1 norm regularizor on topic term. The model can be efficiently solved with gradient method.

Word-Net

In contrast to mining semantics from text corpus, Word-Net is a large English lexical database which organises words in groups (synsets) [115]. Word-Net is notably exploited for the graph structure which provides direct relatedness measurement between words according to the tree structure [116]. The extensiveness and hierarchical structure makes Word-Net suitable for word-sense disambiguation, information retrieval and text classification. However, Word-Net as a

taxonomy, is naturally not able to provide an explicit vectorized representation.

2.3 Video Behaviour Analysis

Video behaviour analysis is a central issue to social interaction and communication [1]. In this section, we briefly review the general tasks of video behaviour analysis involved in this thesis. In particular, we are interested in three sub-categories of behaviour analysis - surveillance behaviour analysis and human action analysis and crowd behaviour analysis.

2.3.1 Video Surveillance Analysis

The widespread use of public space CCTV camera systems has generated unprecedented amounts of data which can easily overwhelm human operators due to the sheer length of the surveillance videos and the large number of surveillance videos captured at different locations concurrently. This has motivated numerous studies into automated means to model, understand, and exploit this data. Some of the key tasks addressed by automated surveillance video analysis include: (1) scene understanding to reveal what are the typical activities and behaviours in the surveilled space [30, 9, 31, 32, 33]; (2) behaviour query by example, allowing the operator to search for similar occurrences to a specified example behaviour [30]; (3) supervised learning to classify/annotate activities or behaviours if events of interest are annotated in a training dataset [9]; (4) video summarisation to give an operator a semantic overview of a long video in a short period of time [34]; and (5) anomaly detection to highlight to an operator the most unusual events in a recording period [30, 9, 31]. In this section, we review each task respectively.

Scene Understanding

Scene understanding is largely based on learning induced semantic representations from visual data. Specifically, given long training video content, one can learn typical activities and behaviours in an unsupervised way. Given object-based motion feature available e.g. object tracking, people model behaviours for example by Hidden Markov Model (HMM) [31, 72], Gaussian Process [117], clustering [118] and stochastic context-free grammars [119]. Given aerial view surveillance scene with sparse behaviour density, one is able to extract reliable tracking results. Due to this reason, we conduct multi-object tracking to extract object-based motion and learn activities as Gaussian Mixture Model in Chapter 3. For crowded and occluded scenes, it is often easier to extract pixel-based motion features such as optical flow. With the pixel-based mo-

tion feature people have widely utilised probabilistic topic model (PTM) for modelling activities [30, 9, 32, 8]. For this reason, we compute optical flow as motion feature and train latent dirichlet allocation (LDA) on bag of words encoded visual feature in Chapter 4 to discover visual topics where each topic corresponds to one type of action.

However, all of these studies operate within-scene rather than modelling globally distributed scenes and discovering shared actions. Despite the benefit of modelling multiple scenes simultaneously, as stated in Section 1.3.1, very few efforts have been dedicated to multi-scene modelling. To recognise the same activity from another viewpoint, Khorkhar et al.[40] proposed a geometric transformation based method to align two events, represented as Gaussian mixtures, before computing their similarity. This model was then used for cross-scene event classification. Nevertheless, the cross-scene matching is designed for a pair of events or actions and is not suitable for modelling multiple scenes at the same time.

In the context of static image (rather than dynamic scene) understanding [120, 121], studies have clustered images by appearance similarity. However, this does not apply directly to surveillance scenes because the background is no longer stationary nor uniform, e.g. building and road appearance, are visually salient but can vary significantly between surveillance scenes at different locations. It is not reliable to relate surveillance scenes based on appearance.

Video Query

Video query has always been an important issue in surveillance applications [122, 30]. Hu et al.[122] used trajectories to learn an activity model and construct semantic indices for video databases. Wang et al.[30] represents video clips as topic profiles and measures similarity between query and candidate clips as relative entropy. Retrieved clips are sorted according to the distance to the query. However none of these techniques take a multi-scene scenario into consideration, where query examples are selected in one scene and candidate clips can be retrieved from other scenes at different locations.

Related to video query, video behaviour annotation/classification has been addressed in the literature [30], also in terms of video segmentation [123]. However, these approaches are typically domain/scene-specific, which means that *each scene* needs extensive annotation of training data; where ideally labels should instead be borrowed from semantically related scenes. Although a recent study [40] recognised events across scenes at the activity level, scene level behaviour classification, and dealing with a heterogeneous database of scenes is still an open problem.

Video Summarisation

Video summarisation has received much attention in the literature in recent years due to the need to digest large quantities of video for efficient review by users. A detailed review can be found in [124]. There are a variety of approaches to summarisation, varying both in how the summary is represented/composed, and how the task is formalised in terms of what type of redundancy should be compressed.

In general, summaries have been composed by: *static keyframes* that represent the summary as a collection of selected key-frames [125], *dynamic skimming* which composes a summary based on a collection of selected clips, and more recently *synopsis*. Synopsis [126, 34] temporally re-orders (spatially non-overlapping) activities from the original video into a temporally compact summary video by shifting activity tubes temporally so they occur more densely. Moreover, the objective of summarisation can be formalised in various ways: to show all foreground activity in the shortest time [126], to minimise the reconstruction error between the summary and the original video, to show at least one example of every typical behaviour, or more abstractly to achieve the highest rating in a user study [125].

As the number of scenes grows, multi-view summarisation becomes increasingly important to help operators monitor activities in numerous scenes. However, multi-view summarisation is much less studied compared to that of single view. Lou et al.[127] adopted multi-view video coding to deal with multi-view video compression, but did not tackle the more challenging compression of semantic redundancy. Fu et al.[128] addressed generating concise multi-view video summaries by multi-objective optimisation for generating representative summary clips. Recently, De Leo et al.[129] proposed a multi-camera video summarisation framework which summarises at the level of activity motif [130]. Due to the severe occlusion, far-field of view and high density activities in surveillance videos, none of the existing techniques solve the problem of distributed multi-scene surveillance video summarisation.

In Chapter 4, we pursue video summarisation from the perspective of selecting the smallest set of representative video clips that still have good **coverage** of all the behaviours in a collection of scenes. Such *multi-scene* summarisation compresses redundancy across scenes as well as within scenes. This corresponds to an application scenario where the user tasked with monitoring a set of cameras wants an overview of all the behaviours that occurred in a set of video streams

during a recording period regardless the source of the video recordings, which typically come from different locations. This perspective on summarisation is attractive because it makes sense of video content indepedent of location and local context. This offers a more holistic conceptual summarisation in a global context as compared to summarisation as visualisation of a single scene in a local context such as video synopsis.

2.3.2 Human Action Analysis

Video human action recognition is now a vast and established area in computer vision and pattern recognition due to the wide application in video surveillance, interaction between human and electronic devices. Extensive surveys of this area are conducted by Aggarwal and Ryoo [3] and Poppe[63]. The general tasks for human action analysis mainly include action recognition and event detection. For action recognition, we summarise as discriminating an action video into a pre-defined set of classes [12, 4, 5]. For event detection, we conclude as given query event category, e.g. 'attempting a board trick', detection algorithm returns a list of videos from the database where related videos are expected to be ranked higher than non-related ones [6, 2]. In this section, we briefly review the most popular human action dataset and event dataset.

Human Action Recognition

Human action recognition has received considerable study in recent years due to the potential application in video surveillance, human computer interaction and video content retrieval. The general task for action recognition is to discriminate action videos into one or more pre-defined categories. The state-of-the-art in action recognition owes to the development of densely tracking points and compute appearance and motion descriptors, namely the dense trajectory [15]. As an emerging learning strategy, deep learning models have been developed for action video recognition and achieve remarkable performance [68]. Since the state-of-the-art video features have been introduced in Section 2.1.1, in this section, we focus on the development of Video datasets for action recognition recognition. We give a summary of human action datasets in Table 2.1. Early datasets focus on simple and isolated human actions performed by a single person, e.g. KTH [12] (2004) and Weizmann [16] (2005) datasets. Both datasets were manually recorded with low-resolution, fixed camera and clean background. Though these videos are far from real world applications, they have inspired continuous research into human action recognition. Due to the growth of internet video sharing, e.g. YouTube and Vimeo, action datasets collection has

shifted from manual recording to online repositories. The more recent human action datasets are mostly collected from YouTube and Vimeo, e.g. OlympicSports [7], HMDB51 [4] and UCF101 [5] and from movies, e.g. Hollywood/Hollywood 2 dataset [17]. These human action videos are closer to realistic applications due to non-static camera, cluttered background, untrimmed videos and multi-person interaction. All these attributes make the recognition task more challenging. Nevertheless, the size of most contemporary action datasets are still limited, e.g. UCF101 with 101 classes and 13K videos. The size of dataset has prevented it from benefiting from deep learning techniques. In order to train deep models for action recognition, Karpathy et al.[67] proposed the Sports-1M dataset with over 1 million action videos which are, however, not manually labelled. Another large-scale human action dataset - Fudan-Columbia Video Dataset (FCVID) [18] was proposed to facilitate deep model training with manually labelled 90K videos.

Table 2.1: A summary of human action datasets

Dataset	#Class	#Videos	Year	Camera	Source
KTH [12]	6	600	2004	Static	Manual Recorded
Weizmann [16]	9	81	2005	Static	Manual Recorded
Hollywood 2 [17]	12	1,787	2009	Dynamic	Movie
OlympicSports [7]	16	800	2010	Dynamic	YouTube
HMDB51 [4]	51	6,766	2011	Dynamic	Movie and YouTube
UCF101 [5]	101	13,320	2012	Dynamic	YouTube
Sports-1M [67]	487	1,133,158	2014	Dynamic	YouTube
FCVID [18]	239	91,223	2015	Dynamic	YouTube

Event Datasets

To recognise more complex events with interactions between people and objects, event datasets including Columbia Consumer Video dataset (CCV) [6] and the TRECVID Multimedia Event Detection (MED) dataset [2] have been developed. In contrast to human action dataset, multimedia event dataset are more temporally unregulated, i.e. event videos are long and unsegmented. Moreover, event detection is often evaluated for ranking performance in contrast to classification in action recognition. Here, event detection refers to video retrieval which returns a ranking of videos in the gallery given a query.

2.3.3 Crowd Behaviour Analysis

Crowd analysis is one of the central topics in computer vision research for surveillance [65]. There are a variety of tasks including: (1) crowd density estimation and person counting [131, 132]; (2) crowd tracking [133, 37]; and (3) crowd behaviour recognition [134, 30, 10]. There are several challenges in crowd behaviour analysis. First of all, one requires both informative and robust visual features from crowd videos. Although simple optical flow [37, 36], tracklets [135, 136], or a combination of motion and static features [8] have been adopted. None of them is both informative and robust. More desirable scene-level features can be further constructed from these low-level features, using probabilistic topic models [30] or Gaussian mixtures [36]. However, these mid-level representations are mostly scene-specific, with a few exceptions such as Khokhar et al.[40] who model multiple scenes to learn a scene-independent representation. Secondly, for recognition in different scenes, existing methods rely heavily upon the assumption of the availability of sufficient observations (a large number of repetitions with variations) from these scenes in order to either learn behaviour models from scratch [30, 8, 36], or inherit models from related scenes [40]. To generalise models across scenes, studies have proposed scene-invariant crowd/group descriptors inspired by socio-psychological and biological research [38], and more recently from deep neural network [10]. In addition to these purpose-built crowd features, dense trajectory features [15] capturing both dynamic (motion boundary) and static textural information have also been adopted for crowd analysis [10]. For learning a scene-invariant model, the method of Shao et al.[10] requires extensive manual annotation of crowd attributes: The *WWW* crowd video dataset [10] has 94 attributes captured by over 10,000 annotated crowd videos, where each crowd video is annotated with multiple attributes. The effort required for annotating these videos is huge. This poses significant challenge to scale up the annotation of any larger video dataset from diverse domains. Thirdly, often the most interesting crowd behaviour is also novel in a given scene/domain. That is, the particular behavioural attribute has not been seen previously in that domain. To address these challenges, in Chapter 7 we explore a different approach to crowd behaviour recognition, by which crowd attribute context is learned from a large body of text descriptions rather than relying on exhaustive visual annotations, and this semantic contextual knowledge is exploited for zero-shot recognition of novel crowd behavioural attributes without labelled training samples.

2.4 Learning Strategies

We review different learning strategies involved in video behaviour analysis. Specifically, we are interested in action recognition in zero-shot scenario, multi-task learning, domain adaptation and multi-label learning.

2.4.1 Zero-Shot Learning

Zero-shot learning aims to achieve dynamic construction of classifiers for novel classes at testing time based on semantic descriptors provided by humans or existing knowledge bases, rather than labelled examples. This approach was popularised by the early studies [137, 138, 19]. Since then numerous studies have been motivated to investigate ZSL due to the scalability barrier of exhaustive annotation for supervised learning, and the desire to emulate the human ability to learn *from description* with few or no examples.

ZSL Architectures

Various architectures have been proposed for zero-shot recognition of novel classes given testing data. Sequential architectures [19, 139, 22, 20, 140, 141, 43] setup visual-semantic classifiers/regressors to predict semantic representations of testing data, followed by a recognition function, e.g. nearest neighbour. The visual-semantic mapping is often learned from training data of known categories and assumed to generalise, and the semantic representation of unknown categories are given by the human or external knowledge. Converging architectures [29, 142, 45, 42] setup energy functions which are positive when visual data and semantic representation is from the same class and negative otherwise. The most widely adopted energy function is a bilinear mapping [29, 45, 42]. Square loss [45] or ranking loss [29, 42] are then employed to learn the mapping. In the testing phase, the energy is computed between testing data and unknown categories and the label for testing data is determined by unknown class with the largest energy score. In Chapter 5 and 6, we adopt a sequential regression approach for simplicity and efficiency of closed-form (rather than the iterative solution of energy-function approaches), and amenability to adaptation to exploiting the unlabelled data manifold.

Attribute Embeddings

The most intuitive intermediate representation for ZSL has been attributes, where categories are specified in terms of a vector of binary [19, 20, 140] or continuous [139, 29, 45] attributes. However, this approach suffers inherently from the need to agree upon a universal attribute ontology,

and the scalability barrier of manually defining each new class in terms of an attribute ontology that grows with breadth of classes considered [139, 29].

Word-Vector Embeddings

While other representations including taxonomic [29], co-occurence [143, 144, 23] and template-based [137] have been considered, word-vector space ZSL [22, 29, 141, 43, 42] has emerged as the most effective unsupervised alternative to attributes. In this approach, the semantic class representation is generated automatically from word-vector embedding which is learned from unstructured text knowledge bases such as the Wikipedia [24]. This can be more intuitively understood as encoding each class name in terms of a vector describing its co-occurance frequency with other terms in a text corpus [141]. In sequential architectures the final recognition is typically performed with nearest neighbour (NN) matching of the predicted class descriptor [141, 43] or the improved graph label propagation [22].

Zero-Shot Learning for Action Recognition

Despite clear appeal from ZSL, few studies have considered it for action recognition. Early attribute-centric studies took latent SVM [20] and topic model [139, 140] approaches, neither of which are very scalable for large video datasets. Thus more recent studies have started to consider unsupervised embeddings including semantic relatedness [143] and word-vectors [26]. However, most prior ZSL action recognition studies do not evaluate against a wide range of realistic set of contemporary action recognition benchmarks, restricting themselves to a single dataset of USAA [139, 140], Olympic Sports [20] or UCF101 [145, 143]. In Chapter 5 and 6, we fully explore word-vector-based zero-shot action recognition, and demonstrate its superiority to attribute-based approaches, despite the latter's supervised ontology construction.

Zero-Shot Learning for Event Detection

In contrast to action recognition, another line of work on the related task of event detection typically deals with temporally longer multimedia videos. The most widely studied test is the TRECVID Multimedia Event Detection (MED) benchmark [2]. In the TRECVID 2013 zero-shot MED task (MED 0EK), 20 events are to be detected among a 27K video (Test Set MED) with no positive examples of each test event available for training. Existing studies [101, 146] typically discover a 'concept space' by extracting frequent terms with pruning in video metadata (per-video text description) and learning concept classifiers on the 10K video Research Set. Then for each of the 20 events to be detected, a query is generated as a concept vector from the metadata of the

event (textual description of event) [101] or an event classifier is learned on 10 positive examples of the testing event [146]. The testing videos are finally tested against the concept classifiers and then matched to the query as inner product between concept detection scores and query concepts [101] or through the event classifier [146]. These approaches rely on two assumptions: (1) a large concept training pool (10K video) with per-video textual description annotated by experts; and (2) a detailed description of the event to be detected is needed to generate the query. For example a typical event description includes the name - 'Birthday Party', Explication - 'A birthday in this context is the anniversary of a person's birth etc', Object/People - 'Decorations, birthday cake, candles, gifts, etc'. Since detailed per-video annotations and detailed descriptions of event types are not widely available in other video databases, in Chapter 5 we focus on exploring the TRECVID task with the more challenging but also more broadly applicable setting of *event name*-driven training and queries only. In another words, we aim to detect complex event by only providing the name, e.g. 'Birthday Party' without naming the Object, Background, etc.

2.4.2 Multi-Task Learning

As establishing the mapping between visual feature and semantic representations is the key to the success of zero-shot action recognition, performance can be further boosted with better generalisable mapping. Multi-Task Learning (MTL) [41] aims to improve generalisation performance of the visual-to-semantic mapping by modelling and exploiting the shared knowledge across the tasks. Various sharing structures have been proposed to model the relations between tasks. An early study [61] proposed to model the weight vector for each task t as a sum of a shared global task $\mathbf{w}_0$ and task specific parameter vector $\mathbf{w}_t$. However, the assumption of a globally shared underlying task is too strong, and risks inducing *negative transfer* [41]. This motivates the Grouping and Overlapping Multi-Task Learning (GOMTL) [62] framework which instead assumes that each task's weight vector is a task-specific combination of a smaller set of latent basis tasks. This constrains the parameters of all tasks to lie on the lower dimensional manifold.

MTL methods have been studied for action recognition [147, 49, 148, 149]. However, all of these studies focus on improving standard *supervised* action recognition with multi-task sharing. For example, considering each of multiple views [148, 149], feature modalities [49], or – most obviously – action categories [147] as different tasks. In Chapter 6, we take a very different approach and treat each dimension of the visual-semantic mapping as a task, in order to leverage MTL to improve source-target generalisation across the disjoint target categories. Finally, we

note that the use of MTL to learn the visual-to-semantic mapping provides a further benefit of a lower-dimensional space in which zero-shot recognition can be better performed due to being more meaningful for NN matching [60].

2.4.3 Domain Adaptation

Domain shift is a widely studied problem in transfer learning [41], although it is usually induced by sampling bias [150, 151] or sensor change [152] rather than the disjoint categories in ZSL. Numerous techniques have been proposed to tackle the domain shift issue. However, most works focus on domain adaptation in standard supervised problems where both the source and target labels are available. This assumption is too demanding for zero-shot scenario where target domain training data is not available or not labelled during the training stage. We, therefore, provide a very initial review on domain adaptation in the context of zero-shot learning.

Importance weighting

Importance weighting (IW) [58, 151] has been one of the main adaptation techniques to address this issue. The idea behind these techniques is to align the source with the target data to maximise performance on the target data. Importantly the prior work in this area is designed for the standard domain transfer problem in a *supervised* learning setting [153], while in Chapter 6 we are the first to generalise it to the *zero-shot* learning scenario. The IW technique we generalise is related to another domain adaptation approach based on discovering a feature mapping to minimise the *Maximum Mean Discrepancy* (MMD) [154, 155] between distributions. However MMD, is less appropriate for us due to focus on feature mapping rather than instance reweighing, and our expectation is that only subsets of auxiliary instances will be relevant to the target rather than the holistic auxiliary set.

Hubness Correction

Hubness is an intrinsic problem in high dimensional data space [46, 156, 157]. In the context of zero-shot learning, hubs are informally defined as the novel categories, referred to as testing prototypes, with high affinity or low distance to many other testing data in the semantic embedding space. With the hubness phenomenon, applying simple nearest neighbour classifier to zero-shot categories would result in many false assigning to hub categories. To mitigate hubness effect, Dinu et al.[46] proposed a simple approach based on the global distribution of testing data. Instead of directly matching testing data to category prototypes, the original distance, e.g.

cosine distance, is replaced by the normalised distance w.r.t. each category prototype or the rank of testing sample w.r.t. each category. With this preprocessing, hub categories can be significantly reduced. Alternative to the ranking correction, Fu et al.[158] proposed a adapting method to adjust the testing category prototypes in accordance with the distribution of testing data. This method is termed as self-training. Specifically, self-training selects the k nearest neighbours (knn) of each category w.r.t. testing data. The selected testing data are averaged to replace the original category prototype. Nearest neighbour matching is then performed with the new prototypes. The selection of parameter K in knn is vital to the success of zero-shot recognition.

Overall, domain adaptation in zero-shot context is still insufficiently studied though has the potential to improve zero-shot learning substantially. Importantly, all of the existing zero-shot domain adaptation techniques are transductive i.e. requiring the access to testing data or categories. This is still a very strong assumption in open world zero-shot recognition problems where there is unlimited testing data and testing categories are not pre-defined.

2.4.4 Multi-Label Learning

Multi-label learning (MLL) studies the problem of learning and associating each instance with multiple labels [53]. The traditional assumption of machine learning problems are mostly focused on single-label learning (SLL) where each sample is associated with only one positive label. Therefore models/classifiers are usually learned separately for each individual category. As opposed to SLL, the multi-label nature encodes the relation between multiple labels and thus enables a way to jointly learn multiple categories to improve learning efficacy. In this section, we firstly briefly review the prevailing ideas for multi-label learning in the conventional supervised learning scenario. Then we give a study of the integration of multi-label learning and zero-shot learning which is an intermediate area largely overlooked by the researchers in MLL and ZSL.

Conventional Multi-Label Learning

MLL [53] is the task of assigning a single instance simultaneously to multiple categories. MLL can be decomposed into a set of independent single-label problems to avoid the complication of label correlation [159, 160]. Although this is computationally efficient, ignoring label correlation produces sub-optimal recognition. Directly tackling the joint multi-label problem through considering all possible label combinations is intractable, as the size of the output space and the required training data grow exponentially w.r.t. the number of unique labels [161]. As a

compromise, tractable solutions to correlated multi-label prediction typically involve considering *pairwise* label correlations [162, 163, 164], e.g. using conditional random fields (CRF)s. However, all existing methods require to learn these pairwise label correlations in advance from the statistics of large labelled datasets. Though this assumption does hold in conventional supervised learning, it would nevertheless fail on zero-shot learning scenario where testing categories are observed during the training stage at all. In the following section, we discuss the development of multi-label prediction for labels without any existing annotated datasets from which to extract co-occurrence statistics, i.e. multi-label zero-shot learning.

Multi-Label Zero-Shot Learning

Although zero-shot learning is now quite a well studied topic, only a few studies have considered multi-label zero-shot learning (MLZSL) [27, 144]. Joint multi-label prediction is challenging because conventional multi-label models require pre-computing the label co-occurrence statistics, which is not available in the ZSL setting. The study given by Fu et al.[27] proposed a Direct Multi-label zero-shot Prediction (DMP) model. This method synthesises a power-set of potential testing label vectors so that visual features projected into this space can be matched against every possible combination of testing labels with simple NN matching. This is analogous to directly considering the jointly multi-label problem, which is intractable due to the size of the label power-set growing exponentially (2^n) with the number of labels being considered. An alternative study was provided by Mensink et al.[144]. Although applicable to the multi-label setting, this method used co-occurrence statistics as the semantic bridge between visual features and class names, rather than jointly predicting multiple-labels that can disambiguate each other. A related problem is to jointly predict multiple attributes when attributes are used as the semantic embedding for ZSL [165]. In this case, the correlations of mid-level attributes, which are multi-labelled, are exploited in order to improve single-label ZSL, rather than the inter-class correlation being exploited to improve multi-label ZSL.

As discussed, MLZSL is still an under explored learning strategy because of the nature of disjoint training and testing categories. The key challenges to the success of MLZSL is to predict the relation between testing categories based on the knowledge of seen training classes. To solve this problem, we propose to encode the relation between categories via a mapping and semantic vectors of class names in Chapter 7. The mapping is learned on known categories and can be transferred to novel/unseen categories to predict the relation. We tested this mechanism on crowd

video attribute prediction and proved better performance than the conventional single-task zero-shot learning strategies.

2.5 Summary

The previous sections have discussed the related works from different aspects. To summarise, we discuss in this section how do the existing works relate to each individual problems addressed in this thesis. In particular, we point out the limitations in the existing approaches compared with our proposed approaches and briefly introduce the solutions in each chapter.

1. (Chapter 3) **Cross-Scene Semantic Behaviour Recognition**: Recognising semantic meaningful behaviours, e.g. traffic going straight, in a view-invariant fashion is of particular interest. Existing work [40] has studied the problem by estimating a homographic transformation to registrate motion events/patterns observed from different view angles. Though a pair of semantic similar events observed from aerial views can be very well aligned the problem of where to transfer is yet solved. In this chapter, we aim to propose a strategy to automatically select the source domain where semantic labels can be transferred to recognise target motion events under a homographic view point change.

2. (Chapter 4) **Semantic Space Discovery for Multi-Scene Behaviour Analysis**: Modelling motion behaviours has been addressed by many existing works [30, 9, 32]. However, the existing studies focus primarily on single scene analysis, i.e. training and testing on the same scene. Other multi-scene models usually assume a topographically relation between cameras [166]. In this chapter, we make no topographic assumptions between cameras but instead analyse a large number of scenes collectively. By grouping scenes into clusters we can discover a shared semantic representation and further achieve a number of multi-scene surveillance tasks.

3. (Chapter 5) **Semantic Space for Zero-Shot Action Recognition**: Recognising human actions from social media videos is more challenging than analysing surveillance videos in that it involves much more behaviour categories, unconstrained background and camera motion. More importantly, the ever increasing number of action categories has lead to very expensive data collection and annotation. The existing supervised action recognition pipeline [15] has shown its weakness to generalise the known knowledge to help recog-

nise new categories. In contrast, we advocate a zero-shot learning strategy to recognise novel action categories via a semantic intermediate space. In particular, we propose to exploit an unsupervised word-vector embedding [24] rather than manually labelled attribute embedding [39] to achieve this task.

4. (Chapter 6) **Multi-Task Semantic Embedding with Prioritised Data Augmentation**: Traditional zero-shot learning model treats the visual-to-semantic mapping as multiple independent regression or classification problems [39, 26]. The correlations between dimensions of word-vector or attributes are largely ignored. Moreover, training data are often given uniform weight by existing models which is not optimal for recognising certain target action categories. In this chapter, we propose to model the visual-to-semantic mapping as a multi-task regression problem in conjunction with selective training data weighting to maximise the ability to generalise to recognising novel categories.

5. (Chapter 7) **Zero-Shot Crowd Behaviour Analysis**: Crowd behaviour analysis has been a key task in surveillance applications. A traditional approach towards crowd behaviour recognition is by annotating crowd attributes and train recognisers [10]. Nevertheless, interesting crowd behaviours are often rare, e.g. violence. Thus it is difficult to collect enough training data for learning a good crowd behaviour recogniser. In this chapter, we propose to extend the zero-shot learning pipeline for crowd behaviour analysis. Importantly, we adapt the conventional zero-shot learning pipeline to multi-label scenario to further improve the efficacy of zero-shot crowd behaviour prediction.

Chapter 3

Cross-Scene Semantic Behaviour Recognition

Existing learning-based outdoor wide-area scene interpretation models suffer from requiring long term data collection in order to acquire statistically sufficient model training samples for every new scene [30, 36, 8]. This makes installation costly, prevents models from being easily relocated. The challenge is how to re-use the knowledge learned from known/source scene to assist the recognition of behaviours in target scenes with minimal effort to collect training data. To tackle this challenge, this chapter first learns semantic action representation from trajectory data. Then we adopt a geometrical matching approach to relate action models learned from a database of source scenes to the target scene with a handful sparsely observed data in a new target scene. This framework is capable of online sparse-shot anomaly detection and semantic action classification in the unseen target scenes, without the need for extensive data collection, labelling and online model training for each new target scene. Crucially, to provide cross-scene interpretation without risk of dramatic negative transfer, we introduce and formulate a scene association criterion to quantify transferability of semantic actions from one scene to another. Extensive experiments show the effectiveness of the proposed framework for cross-scene semantic action classification, anomaly detection and scene association.

In the remainder of this chapter, we first introduce a probabilistic mixture model for object trajectories and trajectory clusters (Section 3.1), followed by a view invariant distance metric for action matching (Section 3.2). Moreover, we also show how to quantify cross-scene transferability and select appropriate source scene for semantic action model transfer (Section 3.2.3). Finally, building on these two capabilities, we present a model for cross-scene sparse-shot anomaly

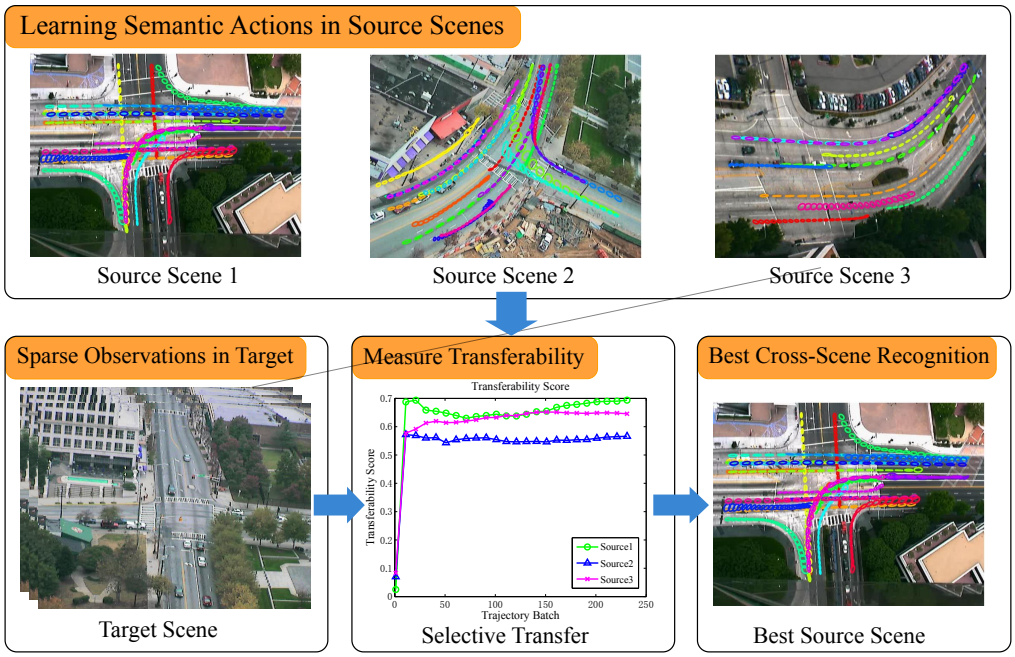


Figure 3.1: Source scene selection procedure: Learning semantic actions in each source scene separately. Then select best source scene via transferability measurement in Eq (3.6).

detection and semantic action classification (Section 3.2.4).

3.1 Semantic Action Representation

We first make a clear definition of semantic space throughout this chapter. As we are trying to model visually coherent and semantic meaningful motion patterns we can define the semantic space as visually similar motion patterns. Examples are illustrated in Figure 3.3. Importantly, we note these motion patterns have to be semantic meaningful under different viewpoints. So the motion patterns in the semantic space are homography-invariant. With this definition, homography-invariant motion patterns are also termed as semantic actions in this chapter. Alternative motion driven semantic actions are studied as well including semantic scene decomposition [8] and trajectory clusters [31]. Among all these alternative definitions, motion pattern is superior in that it can be represented as probablistic models. It makes direct comparison easier and allows geometrical manipulation.

We construct semantic action representation as a probabilistic model from observed object trajectories. Such semantic probabilistic model is often referred to as motion pattern/model in the literature [36, 40]. We track each individual object, e.g., a vehicle in a traffic scene, using a multi-object tracker [167]. Each trajectory $\mathcal{T}$ is represented by a sequence of coordinates and time-stamps $\mathcal{T} = \{(x_i, y_i, t_i)\}$. Note that this representation keeps the directional information via

time-stamp t. We filter broken trajectories with a threshold on minimum length.

3.1.1 Modelling Semantic Action

To learn a domain-specific semantic action in a given scene, we cluster object trajectories to obtain typical motion patterns or events for the scene. Before clustering, we normalise trajectory length. To that end, we employ the Douglas-Peucker algorithm [168] to segment the trajectory at a set of control points. A re-sampled trajectory with a fixed number of points is then obtained by linearly interpolating each interval between proximal control points. Given this pre-processed set of length-normalised object motion trajectories, we over-cluster them using Fuzzy C means (FCM) with Euclidean distance [70] into a large number of C_0 clusters to ensure all modes of object behaviour in the given scene are represented. For each cluster $c = 1,\ldots,C_0$, we fit a N_k component Gaussian mixture model (GMM) to the fixed length trajectories in that cluster. Each trajectory cluster therefore has a probabilistic representation as $g_c(\mathbf{x}) = \sum_{k=1}^{N_k} w_k \mathcal{N}(\mathbf{x}; \mu_{k,c}, \Sigma_{k,c})$ (where μ and Σ are the mean and covariance of each GMM component). To reduce the computational burden, we eliminate redundant clusters from the over-clustering FCM step by computing pairwise KLD (see Section 3.2) between each cluster, and applying self-tuning spectral clustering [169] to determine automatically the optimal number C of motion patterns (trajectory clusters) representing the typical motion events in the scene. This over-clustering followed by pruning process ensures that all the modes of variability in the scene are represented. Without this, direct clustering can result in the most common trajectory types dominating and less-common motion events not being modelled.

3.1.2 Modelling Individual Event

Now, we need to establish a probabilistic representation of an individual run-time object event/trajectory to interpret under the semantic actions defined in the previous section. Here, we define the event as an individual trajectory while the semantic action as an abstracted behaviour learned from a collection of observed trajectories. We define a GMM for each event by a Gaussian centred on each observation $[x_j, y_j, t_j]$ with diagonal covariance. x and y variance are set to the bounding-box size – since the object centre is somewhere in the bounding box – and t variance set to σ_t so to reflect maximum expected speed.

3.2 Cross-Scene Transfer of Semantic Actions and Events

Given the proposed probabilistic representation of semantic actions and events, we now describe a similarity measure to compare semantic action and an event. We first describe the within-scene case before generalising to the across-scene case which needs to account for the potentially different scene geometries.

3.2.1 Within-Scene Comparisons

To quantify the similarity between an event and a semantic action, we exploit the Kullback-Leibler Divergence [40] (KLD) which measures the similarity between probabilistic distributions. The KLD between two distributions $g_m(\mathbf{x})$ and $g_t(\mathbf{x})$ is:

$$\mathcal{KLD}(g_m \,||\, g_t) = \int g_m(\mathbf{x}) \log\left(\frac{g_m(\mathbf{x})}{g_t(\mathbf{x})}\right) \mathrm{d}\mathbf{x}. \tag{3.1}$$

Since there is no analytical solution for the KLD in the case of GMMs distributions, we employ a Monte Carlo approximation [170]. N_m points $\mathbf{x}_m = [x_m, y_m, t_m]$ are sampled from $g_m(\mathbf{x})$ and used to approximate the KLD by evaluating their likelihood under $g_t(\mathbf{x})$:

$$\mathcal{KLD}(g_m \,||\, g_t) \approx \frac{1}{N_m} \sum_{m=1}^{N_m} \log\left(\frac{g_m(\mathbf{x}_m)}{g_t(\mathbf{x}_m)}\right). \tag{3.2}$$

A trajectory cluster typically has larger variance than an individual trajectory, the forward KLD is usually much greater than the backward KLD. To obtain a more stable similarity metric suitable for comparing both trajectories and clusters, we utilise the average of forward and backward measures to define a symmetrical measure for event and semantic action.

$$\mathcal{D}_{\mathcal{KL}}(g_m, g_t) = \frac{1}{2}\left(\mathcal{KLD}(g_m \,||\, g_t) + \mathcal{KLD}(g_t \,||\, g_m)\right) \tag{3.3}$$

3.2.2 Cross-Scene Mapping

We are ultimately interested in cross-scene event and semantic action comparison. For wide area surveillance, semantically equivalent actions differ only in their view geometry, but are equivalent under an geometric similarity transformation $\mathbf{H}$ (a 3×3 matrix). That is, the same, or two semantically equivalent actions/events viewed from differing angles cannot be compared directly unless the translation, scaling and rotation ($\mathbf{H}$) that relates them is known. Therefore we

define a distance measure capable of comparing an event and semantic action across different view-points which should be invariant to the similarity transform $\mathbf{H}$ relating them.

We quantify the view-invariant distance $D^{\mathbf{H}}(g_m, g_t) = \mathcal{KLD}(g_m || g_t^{\mathbf{H}})$ under the similarity transformation $\mathbf{H}$. Here $g_t^{\mathbf{H}}$ indicates geometric transformation of the motion model $g_t(\mathbf{x})$ by $\mathbf{H}$. The optimal transformation $\mathbf{H}^*$ is the one that maximises their similarity. For GMMs models under the distance approximation Eq. (3.2), this corresponds to maximising the likelihood of points sampled from GMM g_m under *transformed* GMM distribution $g_t^{\mathbf{H}}$:

$$\mathbf{H}^* = \underset{\mathbf{H}}{\operatorname{argmax}} \log \prod_{m=1}^{N_m} \sum_{k=1}^{N_k} w_k \mathcal{N}(\mathbf{x}_m; \mathbf{H}\mu_k, \mathbf{H}\Sigma_k\mathbf{H}^T). \tag{3.4}$$

As in Khokhar et al.[40], we approximately optimise Eq. (3.4) by proxy of alternating estimating point correspondences using the Hungarian algorithm [171], and directly fitting $\mathbf{H}$ given fixed correspondences [172] (illustrated in Figure 3.2). This is necessary because without correspondence least-squares transformation estimation (LSE) is meaningless, but correspondence cannot be estimated unless the two patterns are in alignment.

In contrast to Khokhar et al.[40], there are three notable differences: (1) we do not need the path-context information required to avoid the local minima problem in Khokhar et al.[40]. This is because the local minima is in fact a mismatch in temporal order, whereas in our case the temporal information is already modelled by the third (time) dimension of our probabilistic trajectory model; (2) we use the Hungarian-LSE alternation of Khokhar et al.[40] to get an initial condition, followed by direct optimisation for Eq. (3.4) using BFGS [173]. This is a better solution than solely optimising Eq. (3.4) by proxy [40], because there is no formal relation between the alternation and Eq. (3.4); and (3) finally, for our final distance metric between event and semantic action, both within and across scenes, we use the symmetrical KL-Divergence (Eq. (3.5)) whilst the model of Khokhar et al.[40] only performs an unsymmetrical comparison.

$$D^{\mathbf{H}^*}(g_m, g_t) = \frac{1}{2}\left(\mathcal{KLD}(g_m || g_t^{\mathbf{H}^*}) + \mathcal{KLD}(g_t^{\mathbf{H}^*} || g_m)\right) \tag{3.5}$$

The methods described in this section enable cross-scene (similarity transform invariant) comparison of events and actions. However the central issue with exploiting this capability in practice is that these comparisons are only useful / meaningful if the scenes across which they are being compared are semantically related. This is a fundamental question unaddressed by the model of [40]. We shall address this problem next.

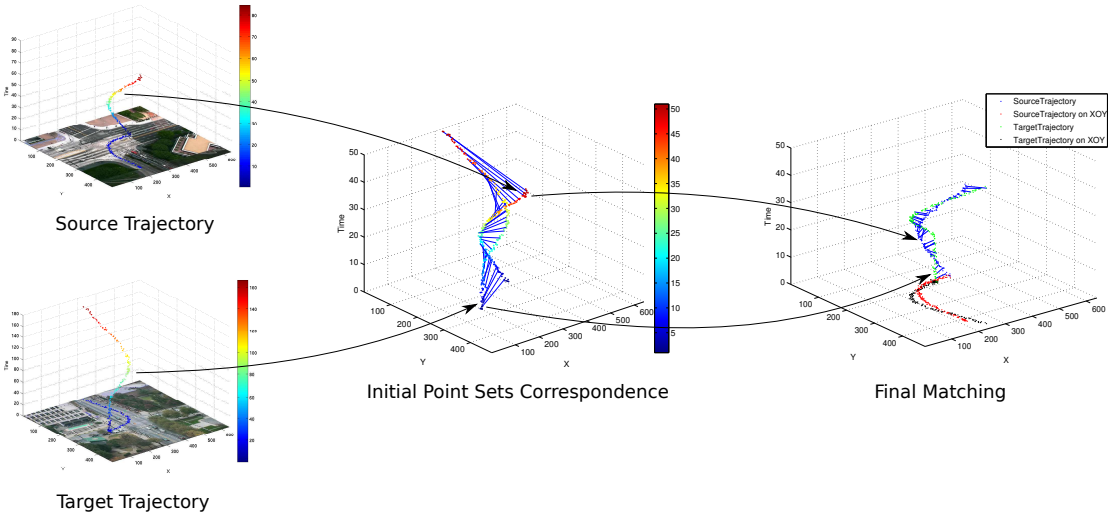


Figure 3.2: Cross-scene trajectory matching process. From left to right, two trajectory point sets are corresponded by the Hungarian algorithm, then a transformation is estimated based on the correspondence. The transformation is obtained by iterating this process.

3.2.3 Transferability Measurement

In this section, we describe how to quantify transferability between domains in order to effectively exploit the cross-scene comparisons introduced in the previous section and hence achieve fully automatic sparse-shot cross-scene classification and anomaly detection. This question of 'from where' to transfer is a known hard problem in transfer learning [41]. Relating one scene to an irrelevant scene typically results in worse performance than no transfer at all (negative transfer), and avoiding this is crucial. Insofar as the 'from where' to transfer problem has been addressed [41], it typically requires labeled data in the target and source scene. Importantly, we will avoid making this assumption here, as relaxing this impractical requirement significantly increases the usefulness of such a system.

Assume there are $s = 1,\ldots,S$ available source scenes. For each of these we have learned a collection of $c = 1,\ldots,C_s$ semantic actions in the form of Gaussian Mixture Models. Now given a target scene t with a set of events $\mathcal{T}^t = \{g_t(\mathbf{x})\}$ observed online, we determine the most relevant source scene s^* for transfer to the target scene t by matching the distribution of event-action distances within the source ($\mathcal{H}_{ss}$) and across the target-source mapping ($\mathcal{H}_{ts}$):

$$\begin{aligned}
\mathcal{H}_{ts} &= \mathcal{H}_{g_i \in \mathcal{T}^t}(\min_{c \in 1\cdots C_s} D^{\mathbf{H}^*}(g_{s,c}, g_i)),\\
\mathcal{H}_{ss} &= \mathcal{H}_{g_j \in \mathcal{T}^s}(\min_{c \in 1\cdots C_s} D(g_{s,c}, g_j)),\\
s^* &= \operatorname{argmin}_s(\|\mathcal{H}_{ts} - \mathcal{H}_{ss}, \|).
\end{aligned} \tag{3.6}$$

Here $\mathcal{H}(\cdot)$ indicates the histogram operator, $g_i \in \mathcal{T}^t$ and $g_j \in \mathcal{T}^s$ index target (g_i) and source (g_j) trajectory events and respectively, $g_{s,c}$ are the learned semantic actions in the source scene, $\mathbf{H}^*$ is the optimal cross-domain transform (Eq. (3.4)), $\|\cdot\|$ is Euclidean distance, and the minimisation over C_s indicates matching semantic actions in source s. Thus scenes are encoded by the *spread* of fits between trajectory events and semantic actions, and matched by the similarity of those spreads.

This is derived from the intuition that two scenes which appear semantically similar to humans should have a similar distribution of motion. Two obvious alternative strategies to source scene selection are: (1) finding a rigid (rather than per-trajectory) similarity transform of all the target trajectories to sources, however this is computationally intractable and non-robust to e.g., piece-wise differences in scene layout; and (2) finding the 'best fit' source with minimum distance of individual target trajectory events to the closest source scene action (Eq. (3.7))

$$s^* = \text{argmin}_s(\sum_{i \in T^t} \min_{c \in 1 \cdots C_s} D^{\mathbf{H}^*}(g_{s,c}, g_i)). \tag{3.7}$$

However this will *over fit* in that a complicated source scene with many different behaviours will always be the best fit for any target scene. In contrast, the proposed method is tractable and does not suffer from over fitting, as considering the full distribution of distances differentiates such domains.

As data is observed online in a target scene, we continually estimate and dynamically select the source domain for transfer via Eq. (3.6). Importantly, as we will show in the experiments, a good source scene can be selected with much less data than is required to build an effective local model in the target scene.

3.2.4 Sparse-Shot Anomaly Detection and Cross-Scene Event Classification

Given the scene-independent distance metric as explained in Section 3.2, and the optimal scene matching procedure as explained in Section 3.2.3, sparse-shot cross-scene event classification and anomaly detection is straightforward as follows. Trajectories represented as $g_t(\mathbf{x})$ in the target scene can be classified using the class c^* of their nearest semantic action:

$$c^* = \arg \min_{c \in 1 \cdots C_s} D^{\mathbf{H}^*}(g_t, g_{s^*,c}), \tag{3.8}$$

where s^* is the optimal source scene as determined by Eq. (3.6) using the data observed so far.

Importantly, this allows classification in the target scene without requiring any annotations.

For anomaly detection, we consider the (similarity invariant) distance of $g_t(\mathbf{x})$ from the nearest cluster in the chosen source domain s^*:

$$D_t = \min_{c \in 1 \cdots C_s} D^{\mathbf{H}^*}(g_t, g_{s^*,c}). \tag{3.9}$$

Anomalous trajectories are flagged as those with distances D_t above a threshold θ_{th}. By quantifying abnormality in relative to selected source scene, significantly better performance can be obtained than by using local action models learned on sparse online observations. This is because the sparse target scene data can be used more effectively to select a source scene rather than to construct a good local model from scratch.

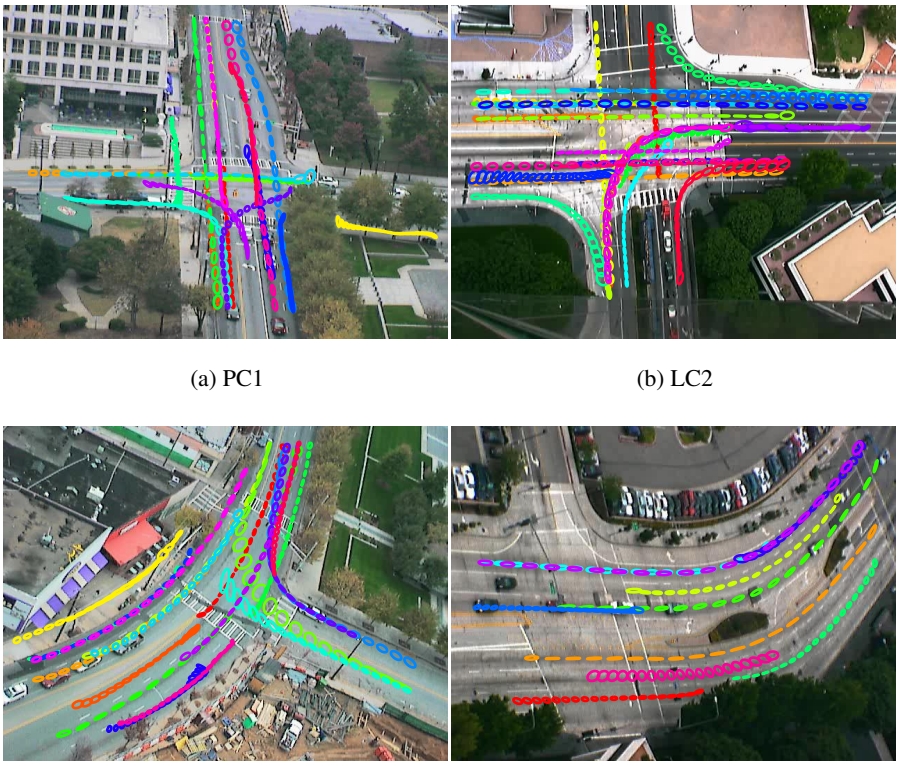
3.3 Experiments

3.3.1 Datasets

The motivating tasks of our framework are anomaly detection and event classification in surveillance videos captured from far-field view. Therefore the NGSIM dataset [174] which is mainly taken by fixed cameras from a far-field of view are good candidates. We evaluate our contributions on four scenes from NGSIM dataset: Lankershim 2 (LC2), Lankershim 4 (LC4), Peachtree 1 (PC1), and Peachtree 3 (PC3). These cover a variety of view angles and scene types, see Table 3.1 and Figure 3.3.

3.3.2 Preprocessing and Settings

For each scene, we extract all available trajectories (Table 3.1). We then over-cluster trajectories (Section 3.1; using $C_0 = 80$ as this is significantly more than the number of typical motion patterns) followed by self-tuning spectral clustering to merge motion models into representative semantic actions (Table 3.1) as illustrated in Figure 3.3. We test the performance of sparse-shot anomaly detection and classification on all 4 scenes in a leave one dataset out protocol, i.e. we evaluate each dataset in turn as a target while considering the other three datasets as source domains.



(a) PC1 (b) LC2

(c) PC3 (d) LC4

Figure 3.3: Learned semantic actions for each scene.

3.3.3 Alternative Models

We compare the following three models: (1) **Direct Transfer** by fixing each source scene in turn as the source; (2) **Local** by building a local model online with limited data (only for anomaly detection, not classification since annotation is assumed unavailable). This is the conventional approach to classification and anomaly detection [31, 70] generalised to online learning. For online learning we process target scene trajectories in chunks and build an updated semantic action model after observing every N additional events/trajectories, using this model to interpret the next chunk of trajectories; (3) **Baseline** by brute-force transfer, aggregating motion models from all available source scenes. This provides a baseline for transfer anomaly detection and classification, but without source selection. Trajectories in the target scene are compared with motion models from all available source scenes. (4) **Best Fit Transfer** is the simplest source scene selecting method. We select the source scene with minimal (transformed) distance of individual target trajectories to source action models (Eq. (3.7)) after each batch of observed trajectories. (5) **Selective Transfer** is our full selective domain-transfer model. After each batch of input, we compute the transferability metric Eq. (3.6), and use the selected source to interpret

Scenes	Frames	Rate	Resolution	View	Anomalies	Number of Trajectories	Learned Actions
PC1	29,918	10 f/s	640x480	$45-60^{\circ}$	1	2,317	19
LC2	21,700	10 f/s	640x480	Aerial	3	2412	28
PC3	29,918	10 f/s	640x480	$45-60^{\circ}$	1	1,468	19
LC4	20,950	10 f/s	640x480	$45-60^{\circ}$	3	2,444	10

Table 3.1: Statistics and pre-processing results of each scene (domain).

observed trajectories.

3.3.4 Evaluation and Results

In these experiments, we evaluate the ability of our framework to select source scenes, classify and detect abnormal events. Since there is no clear ground-truth for scene selection, we evaluate source scene selection by way of whether the selected source provides effective classification and detection. For anomaly detection, we manually annotated abnormal events in each scene, as illustrated in Figure 3.4 which includes events such as pedestrian jaywalking, u-turns and swerving. Good models should rank anomalies higher than normal events. To evaluate anomaly detection performance, we therefore compute the receiver operating characteristic (ROC) curve, which reflects true positive versus false positive rate as detection threshold is varied. This is then summarised by area under the curve (AUC) metric. For classification, we manually labelled events in each scene into three categories (turn-left, turn-right, go-straight). Classification performance is then evaluated by simple accuracy for each scene.

Source Scene Selection for Classification

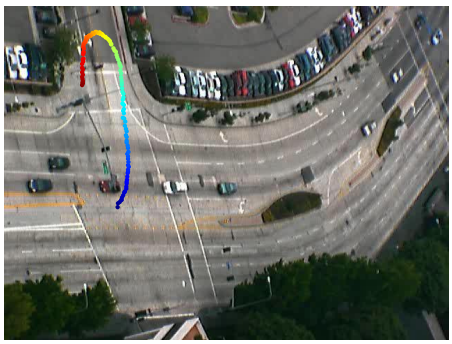
The results for direct and Selective Transfer methods are summarised in Figure 3.5. The first and second column show the source selection transferability metric for **Best Fit** and **Selective Transfer** respectively as a function of observed trajectory batch in the target scene. In each case, both source-selection metrics converge to a consistent source selection. For instance, taking PC1 as the target scene, both the Best Fit and Selective Transfer strategies converge to selecting LC2 as best source (green line higher than others). However as expected, the Best Fit metric consistently prefers the most complex dataset (LC2), whereas only Selective Transfer metric selects a different source in each case, showing the selectivity of our metric.

The third column shows the classification accuracy for each approach (three direct trans-

(a) PC1 (LC2 selected)

(b) LC2 (PC1 selected)

(c) PC3 (LC4 selected)



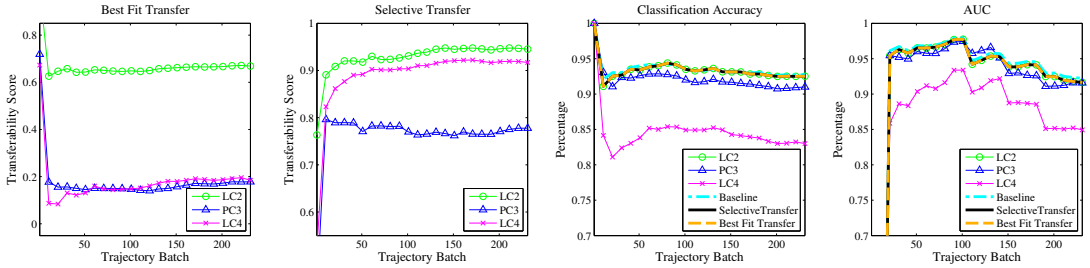
(d) LC4 (PC3 selected)

Figure 3.4: Illustration of abnormal events spotted in each scene. Colour indicates time.

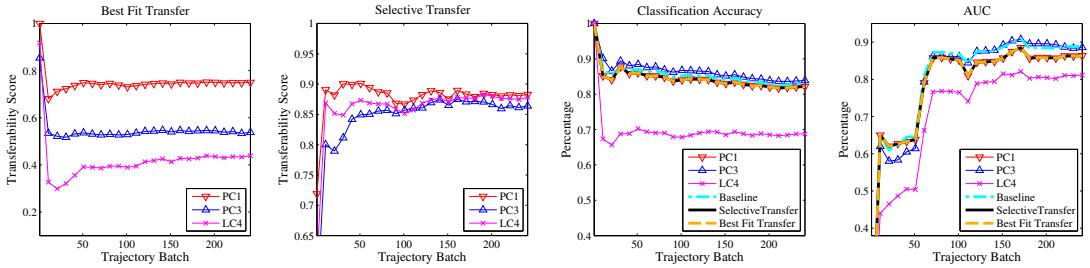
fer conditions with coloured symbols, brute-force transfer (Baseline) in cyan dash-dot, Best Fit Transfer in orange dash and our Selective Transfer framework in bold black). Considering the direct transfer conditions, some source dataset is worst by a significant margin due to the semantic mismatch between the events and actions in source and target scenes. Meanwhile the brute-force baseline and Best Fit Transfer mechanisms are also worst or near worst in some cases due to in favour of most complex source scene. In contrast, across the diverse combinations of sources and targets, our Selective Transfer framework is usually best (PC1) or near best (PC3, LC2 and LC4) overall. Importantly, our Selective Transfer metric consistently avoids the worst source (unlike Best Fit for LC4 and PC3), and is robust in the case where the brute-force baseline is seriously poor (LC4). These results reflect both the serious risk of negative transfer and our framework's robustness to it.

Source Selection for Anomaly Detection

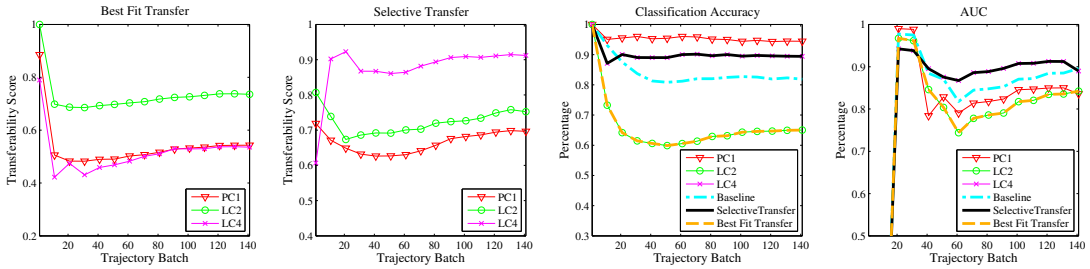
Anomaly detection performance is shown in Figure 3.5, fourth column. Again, in each case our Selective Transfer framework is best (PC3) or near-best (PC1, LC2, LC4) compared to Direct



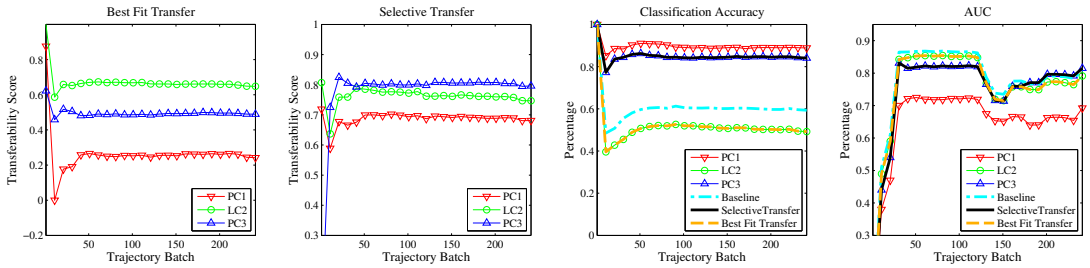
(a) PC1



(b) LC2



(c) PC3



(d) LC4

Figure 3.5: Cross-domain scene understanding. Rows: Target scenes. Columns: (1st) Source selection metric. (2nd) Classification accuracy. (3rd) Anomaly detection AUC.

Transfer. Here the brute-force baseline performs closer to Selective Transfer, but an noticeable margin is still present in the PC3 case, where Best Fit also selects the worst source. PC1 and LC2 are estimated to exhibit mutual transferability as well as PC3 and LC4. This is understandable given the straight nature of the first two scenes and the curvy nature of the latter two scenes. This source selection allows more effective anomaly detection. For example the U-turn and swerving

driving in PC1 and LC2 respectively are ranked more highly as anomalies with LC2 and PC1 as the respective sources than they would be with PC3 and LC4 (which are intrinsically more curvy) as the sources.

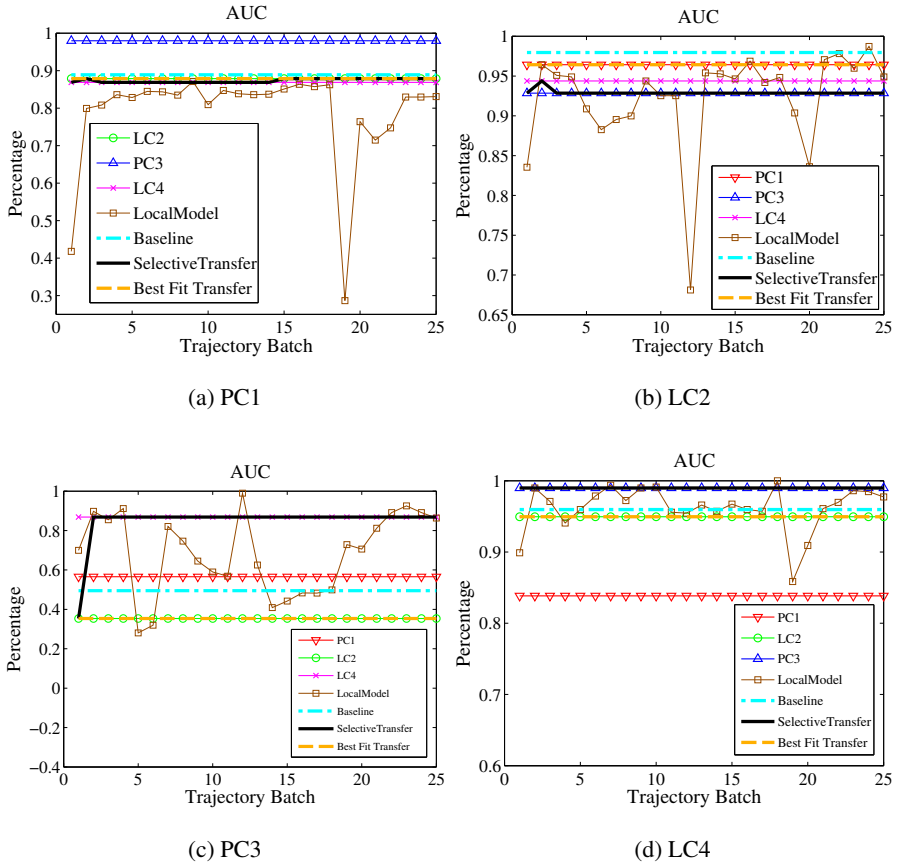
Sparse Data Stability of Source Selection

We next ask how stable is source selection in the rapid deployment / very sparse target data context of interest, and how this compares to building a local model online. To test this we evaluate the detection of each target domain anomaly, embedded in a test set consisting of the 100 adjacent typical trajectories. We vary the size of a learning window (from 1 to 250 trajectories in batches of 10) ahead of test set in the data stream – effectively controlling how much data the local model has to learn typical behaviours, and how much data the transfer model has to select a suitable source scene. The results in Figure 3.6 show that our Selective Transfer framework (bold black) performs reasonably despite the extremely sparse data, selecting the best source in the 2 cases where the margin between best and worst is significant (PC3 and LC4). We note that in the two cases where selective does not make the best choice, it will eventually do so given enough data (PC1 and LC2 in Figure 3.5). Compared to brute-force and Best Fit Transfer, Selective Transfer is in each case same or better in 3 datasets and worse in only one.

The most conventional strategy of building a local model online (brown) generally performs poorly, and importantly is very unstable. This is because with statistically insufficient training data, the rank of the anomaly varies dramatically as the particular samples included in the growing training set vary. Importantly, and in contrast to this instability, the source selection is quite stable even with such sparse inputs – performing consistently from as few as 10 observed trajectories. This highlights the important conclusion that sparse target domain data is much more effectively used for computing Selective Transfer to well understood domains than for building a weak statistically insufficient local model.

Discussion

It is worth noting that a key aim of Selective Transfer is to avoid negative transfer by selecting source models which are suitable for interpreting target events. Previous methods such as Khokhar et al.[40] have considered transfer from one scene to another, but not how to deal with multiple sources of varying relevance. If it used one specific source, then it would roughly correspond to our single source conditions (aside from our technical improvements, mentioned in



(a) PC1 (b) LC2

(c) PC3 (d) LC4

Figure 3.6: Anomaly detection with sparse data: Comparing constructing a local model with sparse data against using this data for domain selection.

Section 3.2). If it aggregated all the source data together, it would roughly correspond to our brute-force baseline condition.

We have seen that Best Fit Transfer falls down in non-selectively preferring the most complex scene. Meanwhile, brute-force transfer is intrinsically limited in the long term, as aggregating multiple sources increases over-fitting monotonically. Consider the case of anomaly detection: as many more source scenes are added to the pool, eventually some scene in which every behaviour is normal has been added. Now every target domain track – even abnormal ones – can be well explained by some source domain data, and anomaly detection is poor. Clearly this misses the point of context: abnormality is context dependent according to the semantics of the scene. Correctly determining the context in which an event should be interpreted is exactly what is achieved by our Selective Transfer mechanism.

3.4 Summary

In this chapter, we propose a novel framework for cross-scene traffic behaviour analysis via semantic action transfer. By learning models in a couple of source scenes offline, we achieve cross-scene sparse-shot anomaly detection and classification in a new scene. Crucially, we introduce a robust scene similarity criterion - Transferability Measurement - that enables robust scene-transfer by finding the most relevant source scene among a collection of scenes. Selecting a well learned source scene turns out to be a much more effective use of sparse local data in target scene than learning a local model. These results are an important contribution toward the topical goals of achieving re-locatable and hence scalable surveillance models.

In this chapter, we define semantic space as a collection of visually similar homography-invariant motion patterns. This implies semantic similar actions/events are subject to a homographic transformation. This assumption prevents from modelling more complex semantic meanings which could be far beyond motion trajectories, e.g. pedestrian crossing zebra line and motorbikes v.s. cars. We believe a integration of motion and appearance model could potentially yield a more fine-grained semantic definition.

Moreover, learning semantic action models (GMM) from trajectories is limited to sparse scenes with little occlusion as the NGSIM dataset where tracking is reliable. However, real surveillance cameras are often installed in a close-to-ground position, it is more likely to expect highly occluded views with very dense traffic. Tracking individual vehicles and pedestrian are thus extremely unreliable. Moreover, analysing multiple scenes at the same time is more interesting than transferring between a pair of scenes due to the large-scale surveillance network. In the following chapter, we propose to extract pixel-based motion feature to account for the challenge of real surveillance scenes and cluster multiple scenes by semantic relatedness to facilitate multi-scene behaviour analysis.

Chapter 4

Semantic Space Discovery for Multi-Scene Behaviour Analysis

As discussed in Section 1.3.1, the growing rate of public space CCTV installations has generated a need for automated methods for exploiting video surveillance data including scene understanding, query, behaviour annotation and summarisation. For this reason, extensive research has been performed on surveillance scene understanding and analysis.

The preceding chapter proposes an approach to compare events and semantic actions across scenes and further compute transferability between scenes to facilitate cross-scene event classification and anomaly detection. Whilst good performance has been achieved for event recognition in the target scene, the cross-scene recognition model is not suitable for analysing multiple surveillance scenes collectively. Because cross-scene model is only able to transfer knowledge between a pair of scenes and more complex interactions and activities are the main concern for surveillance scenes which can not be trivially modelled by the semantic action models proposed in the previous section. Moreover, the semantic similarity between different but related scenes, e.g., many different traffic scenes of similar layout, is not generally exploited to improve any automated surveillance tasks and reduce manual effort. Exploiting commonality, and sharing any supervised annotations, between different scenes is challenging due to: Some scenes are totally un-related and thus any information sharing between them would be detrimental; while others may only share a subset of common actions and thus information sharing is only useful if it is selective. Moreover, semantically similar actions which should be modelled together and

shared across scenes may have quite different pixel-level appearance in each scene. Thus a unified framework which selectively learns semantic space within which data from different scenes are comparable is wanted.

To this end, this chapter proposes a new framework for distributed multiple-scene global understanding that clusters surveillance scenes by their ability to explain each others behaviours; and further discovers which subset of actions are shared versus scene-specific within each cluster. A diagram of the proposed framework is shown in Figure 4.1. We show how to use this structured representation of multiple scenes to improve common surveillance tasks including scene activity understanding, cross- scene query-by-example, behaviour classification with reduced supervised labelling requirements, and video summarisation. In each case we demonstrate how our multi-scene model improves on a collection of standard single scene models and a flat model of all scenes.

In the remainder of this chapter, we first introduce how to learn local semantic action models from each individual scene in Section 4.1. Then we explain the multi-layer action and scene clustering strategy to discover a group of semantic related scenes and a shared action models shared by the related scenes in Section 4.2. This is followed by a brief presentation of cross-scene query by example and classification and multi-scene summarisation. Finally, we conduct extensive experiments on a newly collected multi-scene surveillance video dataset in Section 4.5.

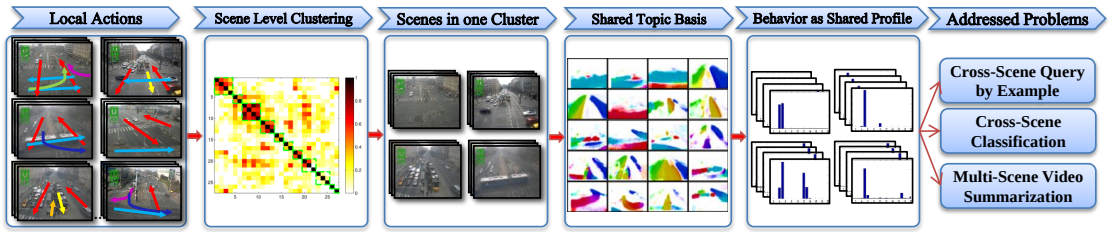


Figure 4.1: An illustration of the proposed framework for multi-scene behaviour analysis.

To make a distinction on semantic space definition from the previous chapter, we clarify in this chapter that semantic actions are defined as temporally coherent motion patterns, e.g. vertical traffic flow with vehicles going up and down simultaneously in Figure 4.3. As multiple scenes are analysed in a collective manner, we model shared semantic actions, which is termed as Shared Topic Basis in Figure 4.1, across a cluster of related scenes. The learned shared actions constitute a shared semantic space where motion event can be represented as shared profile (Behaviour as Shared Profile in Figure 4.1) for further comparison. In the following section, we first introduce

how to construct semantic space in a local scene.

4.1 Learning Local Scene Semantic Actions

Given a set of surveillance scenes we first learn local semantic actions in each individual scene using *Latent Dirichlet Allocation* (LDA) [110]. Without loss of generality, the LDA model studied in this chapter could easily be replaced by more elaborate topic models, e.g. HDP [30]. LDA generates a set of topics to explain each scene. Topics are usually spatially and temporally constrained sub-volumes reflecting the action of a single or small group of objects. Following Wang et al.[30] and Hospedales et al.[9], we use actions to refer to topics and activities/interactions to refer to scene-level state defined by the coordinated actions of all scene participants.

4.1.1 Video Clip Representation

We follow the general approach [30] to construct visual features for topic models. For each video out of an M scene dataset we first divide the video frame into $N_a \times N_b$ cells with each cell covering $N_c \times N_c$ pixels. Within each cell we compute optical flow [83], taking the mean flow as the motion vector in that cell. Then we quantize motion vector into N_m fixed directions. Note, stationary foreground objects can be readily added as another cell state as described in Hospedales et al.[9] and Varadarajan and Odobez [175]. Therefore a codebook $\mathbf{V}$ of size $N_v = N_a \times N_b \times N_m$ is generated by mapping motion vectors to discrete visual words (from 1 to N_v). N_d visual documents $\mathbf{X} = \{\mathbf{x}_j\}_{j=1}^{N_d}$ are then constructed by segmenting the video into non-overlapping clips of fixed length, where each clip $\mathbf{x}_j = \{x_{ij}\}_{i=1}^{N_j}$ has N_j visual words x_{ij}. Clip and document are used interchangeably here with both indicating visual words accumulated in a temporal segment.

4.1.2 Learning Local Actions with Topic Model

Learning LDA for scene s discovers the dynamic 'appearance' of $k = 1 \ldots K$ typical topics/actions (multinomial parameter $\boldsymbol{\beta}_k^s$), and explains each visual word x_{ij}^s in each clip $\mathbf{x}_j^s$ by a latent topic y_{ij}^s specifying which action generated it, as shown in Figure 4.2. The topic selection y_{ij}^s is drawn from multinomial mixture of topics parametrized by $\boldsymbol{\theta}_j^s$ which is further governed by a Dirchelet distribution with parameter $\boldsymbol{\alpha}^s$. In scene s the joint probability of N_d visual documents $\mathbf{X}^s = \{\mathbf{x}_j^s\}_{j=1}^{N_d}$, topic selection $\mathbf{Y}^s = \{\mathbf{y}_j^s\}_{j=1}^{N_d}$ and topic mixture $\boldsymbol{\theta}^s = \{\boldsymbol{\theta}_j^s\}_{j=1}^{N_d}$ given hyperparameters $\boldsymbol{\alpha}^s$ and $\boldsymbol{\beta}^s$ is:

$$p(\boldsymbol{\theta}^s, \mathbf{Y}^s, \mathbf{X}^s \mid \boldsymbol{\alpha}^s, \boldsymbol{\beta}^s) = \prod_{j=1}^{N_d} p(\boldsymbol{\theta}_j^s \mid \boldsymbol{\alpha}^s) \cdot \prod_{i=1}^{N_j} p(y_{ij}^s \mid \boldsymbol{\theta}_j) p(x_{ij}^s \mid y_{ij}^s, \boldsymbol{\beta}^s) \tag{4.1}$$

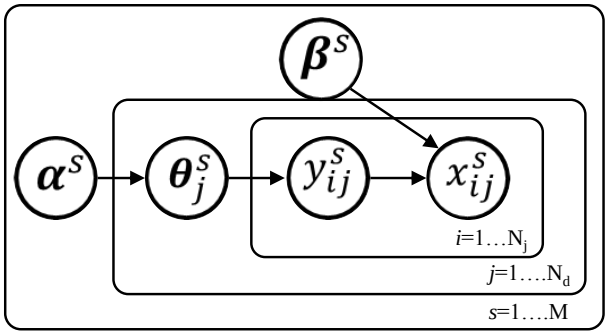


Figure 4.2: Graphical model for Latent Dirichlet Allocation.

Model Inference

Exact inference in LDA is intractable due to the coupling between $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ [110]. Variational inference approximates a lower bound of log likelihood by introducing variational parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\phi}$. Dirichlet parameter $\boldsymbol{\gamma}_j$ is a clip-level topic profile and specifies the mixture ratio of each action $\boldsymbol{\beta}_k$ in a clip $\mathbf{x}_j$. Thus, each video clip is represented as a mixture of actions ($\boldsymbol{\gamma}_j$). The variational EM procedure for LDA is given in Algorithm 1 where $\mathbb{1}(\cdot)$ is an indicator function and $\Psi(\cdot)$ is the first derivative of the $\log\Gamma$ function. For efficiency, we apply the sparse updates identified in Fu et al.[139] for an order of magnitude speed increase.

After learning all $s = 1 \ldots M$ scenes, every clip $\mathbf{x}_j^s$ is now represented as a topic profile $\boldsymbol{\gamma}_j^s$; and each scene is now represented by its constituent actions $\boldsymbol{\beta}_k^s$ (Figure 4.3).

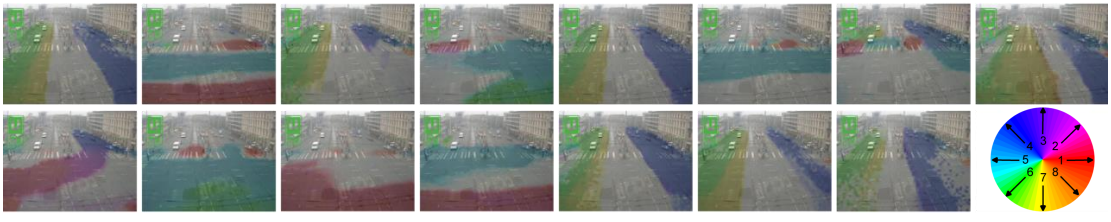


Figure 4.3: Locally learned actions/topics in an example scene. The optical flow is quantized into $N_m = 8$ directions as shown in the colorwheel.

Algorithm 1 Topic model learning for a single scene.

initialize $\boldsymbol{\alpha}_k = 1$

initialize $\boldsymbol{\beta} = random(N_v, K)$

initialize $\boldsymbol{\phi}_{ijk} = 1/K$

repeat

E-Step:

for $j = 1 \rightarrow N_d$ **do**

for $k = 1 \rightarrow K$ **do**

$\boldsymbol{\gamma}_{jk} = \boldsymbol{\alpha}_k + \sum_{i=1}^{N_j} \boldsymbol{\phi}_{ijk}$

for $i = 1 \rightarrow N_j$ **do**

$\boldsymbol{\phi}_{ijk} = \boldsymbol{\beta}_{x_{ijk}} \exp(\Psi(\boldsymbol{\gamma}_{jk}))$

end for

end for

end for

M-Step:

for $v = 1 \rightarrow N_v$ **do**

for $k = 1 \rightarrow K$ **do**

$\boldsymbol{\beta}_{vk} = \sum_{j=1}^{N_d} \sum_{i=1}^{N_j} \boldsymbol{\phi}_{ijk} \mathbb{1}(x_{ij} = v)$

end for

end for

until Converge

4.2 Multi-Layer Action and Scene Clustering

We next address how to discover related scenes and learn shared topics/actions across scenes. This multi-layer process is illustrated in Figure 4.4 for two typical clusters 3 & 7: At the scene level we group related scenes according to action correspondence (Section 4.2.1); within each scene cluster we further compute a *shared action topic basis* so that all actions within that cluster are expressed in terms of the same set of topics (Section 4.2.2).

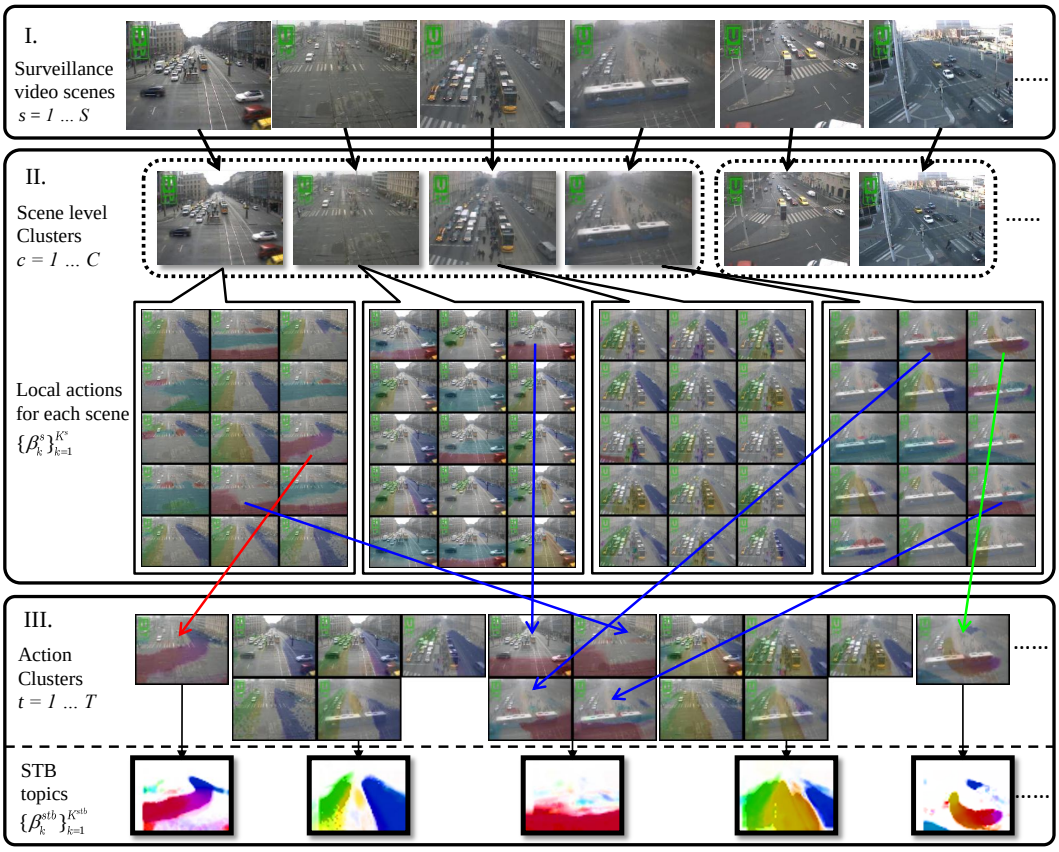


Figure 4.4: An illustration of multi-layer clustering of scenes and actions. Block I (Top) illustrates the original surveillance video scenes. Block II (middle) illustrates (1) related scenes are grouped into clusters (indicated by dashed boxes); and (2) the local topics/actions learned in each scene. Block III (bottom) illustrates (1) local topics are furthered grouped into action clusters (colour lines indicate some examples); and (2) action clusters are merged to construct a *shared action topic basis (STB)*.

4.2.1 Scene Level Clustering

In order to group related scenes, we first need to define a relatedness metric. Related scenes should have more common actions so that the model learned from them is compact. So we assume the scenes with semantically similar actions are more likely to be mutually related. We thus define the relatedness between two (aligned) scenes a and b, by the correspondence of their semantic actions.

Alignment

Comparing scenes directly suffers from cross-scene variance due to view angle. To reduce this cross-scene variance we first align two scenes with a geometrical transformation including scaling h_s and translation $[h_x, h_y]$. Although this is not a strong transform as we assumed in the previous chapter, it is valid in the typical case that a camera is installed upright, and with surveillance cameras there are classic views which can be simply aligned by scaling and translation. To

achieve this, we first denote the transform matrix for normalising visual words in each scene a and b to the origin as $\mathbf{H}^a_{norm}$ and $\mathbf{H}^b_{norm}$ defined as Eq. (4.2). Scaling (h^a_s) and translation (h^a_x, h^a_y) parameters are estimated by Eq. (4.3).

$$\mathbf{H}^a_{norm} = \begin{bmatrix} h^a_s & 0 & h^a_x \\ 0 & h^a_s & h^a_y \\ 0 & 0 & 1 \end{bmatrix} \tag{4.2}$$

$$\begin{aligned} centre &= \frac{1}{N_d \cdot N_j} \sum_{j=1}^{N_d} \sum_{i=1}^{N_j} x^a_{ij}, \\ h^a_s &= \frac{N_d \cdot N_j}{\sum_{j=1}^{N_d} \sum_{i=1}^{N_j} \|x^a_{ij} - centre\|_2}, \\ \begin{bmatrix} h^a_x \\ h^a_y \end{bmatrix} &= -h^a_s \cdot centre \end{aligned} \tag{4.3}$$

Two scenes can thus be aligned by transforming data from a to b via $\mathbf{H}^{a2b} = \mathbf{H}^{b-1}_{norm} \cdot \mathbf{H}^a_{norm}$. So any topic k in a can be aligned for comparison with those in b by $\mathbf{H}^{a2b}$. We denote the topic transformation procedure as $\boldsymbol{\beta}' = \mathbb{H}(\boldsymbol{\beta}; \mathbf{H})$.

Affinity and clustering

Given the scene alignment above, we define the relatedness between scenes a and b by the percentage of corresponding topic pairs. More specifically, given K^a local topics $\{\boldsymbol{\beta}^a_{k^a}\}_{k_a=1\cdots K^a}$ in scene a and K^b local topics $\{\boldsymbol{\beta}^b_{k^b}\}_{k_b=1\cdots K^b}$ in scene b, the distance between topic $\boldsymbol{\beta}^a_{k^a}$ and topic $\boldsymbol{\beta}^b_{k^b}$ is defined as the symmetrical Kullback-Leibler Divergence (KLD) $\mathcal{D}_{\mathbf{KL}}$ in Eq. (4.4):

$$\begin{aligned} \mathcal{D}_{\mathbf{KL}}(\boldsymbol{\beta}^a_{k^a}, \boldsymbol{\beta}^b_{k^b}) &= \frac{1}{2}(\mathcal{KLD}(\boldsymbol{\beta}^{a2b}_{k^a} \,||\, \boldsymbol{\beta}^b_{k^b}) + \mathcal{KLD}(\boldsymbol{\beta}^{b2a}_{k^b} \,||\, \boldsymbol{\beta}^a_{k^a})) \\ \mathcal{KLD}(\boldsymbol{\beta}^a_{k^a} \,||\, \boldsymbol{\beta}^b_{k^b}) &= \frac{1}{N_v} \sum_{v=1}^{N_v} \boldsymbol{\beta}^a_{k^a v} \cdot \log\left(\frac{\boldsymbol{\beta}^a_{k^a v}}{\boldsymbol{\beta}^b_{k^b v}}\right) \end{aligned} \tag{4.4}$$

Given a threshold τ the similarity between two topics can be binarized. Topic pairs with distance less than a threshold are counted as inliers, defined by:

$$NumInlier = \sum_{k^a} \mathbb{1}(\min_{k^b}(\mathcal{D}_{\mathbf{KL}}(\boldsymbol{\beta}^a_{k^a}, \boldsymbol{\beta}^b_{k^b})) < \tau) + \sum_{k^b} \mathbb{1}(\min_{k^a}(\mathcal{D}_{\mathbf{KL}}(\boldsymbol{\beta}^b_{k^b}, \boldsymbol{\beta}^a_{k^a})) < \tau) \tag{4.5}$$

where $\mathbb{1}(\cdot)$ is the indicator function. The final relatedness measure $\mathcal{D}(a,b)$ between scenes a and b is the percentage of inlier topic pairs:

$$\mathcal{D}(a,b) = \frac{NumInlier}{K^a + K^b} \tag{4.6}$$

Since Eqs. 4.4 and 4.5 are symmetric, Eq. 4.6 is as well. Given this relatedness measure, every scene pair is compared to generate an affinity matrix, and self-tuning spectral clustering [169] is used to group scenes into C semantically similar scene-level clusters. (See Figure 4.4 II for an example).

Alternative measurement for scene relatedness have been studied by Xiao et al.[176] and Zhou et al.[177]. In both the SUN dataset [176] and the newer Places Dataset [177], outdoor traffic scenes are present, e.g. 'roundabout' in SUN and 'freeway' in Places. The relatedness between scenes can be measured by distance metrics in feature space. In particular, hand-crafted features including the GIST features [178], HOG [80], Dense SIFT [179], etc., have been studied as well as more recent deep features [177]. Although scene classification with great variance in appearance can be achieved by the appearance feature based approaches, we believe they can barely generalise to the task we are interested in. This is due to both the strict requirement on the match of geometrical layout of scenes in measuring the relatedness and more importantly the match of motion patterns across scenes. The latter demand can be only met by analysing the motion from video sequence rather than from a single still images. In a more likely case, the visual appearance based measurement would overly generalise the layout of scenes and mix all traffic scenes together.

4.2.2 Learning A Shared Action Topic Basis

Scenes clustered according to Section 4.2.1 are semantically similar, however the representation in each is still distinct. We next show how to establish a shared representation for every scene in a particular cluster. We denote the set of scenes in a cluster as $\mathcal{C}$. We first choose the scene with the lowest distance to all other scenes in the cluster as the reference scene/coordinate s_{ref}. Actions in all scenes $s \in \mathcal{C}$ can be projected to the reference coordinates via transform $\mathbf{H}^{s2s_{ref}}$ as stated in Eq. (4.7).

$$\forall s \in \mathcal{C}, \forall k = 1 \dots K : \tilde{\boldsymbol{\beta}}_k^s = \mathbb{H}(\boldsymbol{\beta}_k^s; \mathbf{H}^{s2s_{ref}}) \tag{4.7}$$

Once every topic is in the same coordinate system, we create an affinity matrix for all the transformed topics $\{\tilde{\boldsymbol{\beta}}_k^s\}_{s\in\mathcal{C}}$ using the symmetrical Kullbeck-Leibler Divergence as distance metric (Eq. (4.4)). Hierarchical clustering is then applied to group the projected actions into K^{stb}

clusters $\{\mathcal{T}_k\}_{k=1}^{K^{stb}}$. ($\mathcal{T}_k$ denotes the set of actions in a cluster k). The result is that semantically corresponding actions across scenes are now grouped into the same cluster. We then take the mean of actions in each action cluster $\mathcal{T}_k$ as one *shared action topic* $\boldsymbol{\beta}_k^{stb}$ as in Eq. (4.8). An alternative to this approach is to re-learn topics from the concatenation of visual words of all the scenes in a single cluster. However, this 'Learning-from-Scratch' strategy prevents explicitly identifying shared and unique topics across scenes. Because the trace of local topics from individual scenes to *shared action topic basis* (STB) is lost. In contrast, our framework reveals how scenes are similar or different.

$$\forall k = 1 \ldots K^{stb} : \boldsymbol{\beta}_k^{stb} = \frac{1}{\mid \mathcal{T}_k \mid} \sum_{k',s' \in \mathcal{T}_k} \tilde{\boldsymbol{\beta}}_{k'}^{s'} \tag{4.8}$$

We denote the set of *shared action topics* $\{\boldsymbol{\beta}_k^{stb}\}_{k=1}^{K^{stb}}$ learned for the cluster as the STB. The resulting STB captures both common and unique actions in every scene member, see Figure 4.4 III for an example. We can now represent the behaviours in every scene as STB profiles: by projecting the STB back to each scene and re-computing the topic profile $\boldsymbol{\gamma}_j^{stb}$ defined now on $\{\boldsymbol{\beta}_k^{stb}\}_{k=1}^{K^{stb}}$; in contrast to the original scene-specific representation ($\boldsymbol{\gamma}_j^{s}$, defined in terms of $\{\boldsymbol{\beta}_k^{s}\}_{k=1}^{K}$). That is, re-running Algorithm 1, but with $\boldsymbol{\beta}$ fixed to the STB values obtained from Eq. (4.8). An example of behaviour profiling on STB is illustrated in Figure 4.5. Visual words accumulated within a clip are profiled according to the STB. Thus each behaviour can be treated as a weighted mixture of multiple actions.

4.3 Cross-Scene Query by Example and Classification

Given the structured multi-scene model introduced in the previous section, we can now describe how cross-scene query and classification can be achieved.

4.3.1 Cross-Scene Query

Action-based query by example aims at retrieving semantically similar clips to a given query clip. In the cross-scene context, the pool of potential clips to be searched for retrieval includes clips from every camera in a scene cluster. Within a scene cluster $\mathcal{C}$, we segment each video s into $j = 1 \ldots N_d$ short clips (Section 4.1.1). We represent the jth video clip in scene s as topic profile $\boldsymbol{\gamma}_{js}^{stb}$ defined on STB. A query clip q, represented by STB profile $\boldsymbol{\gamma}_{qs}^{stb}$ can now be directly compared against all other clips in the cluster $\{\boldsymbol{\gamma}_{js'}^{stb}\}_{j,s' \in \mathcal{C}}$ using L2 distance. In this way, *cross-scene*

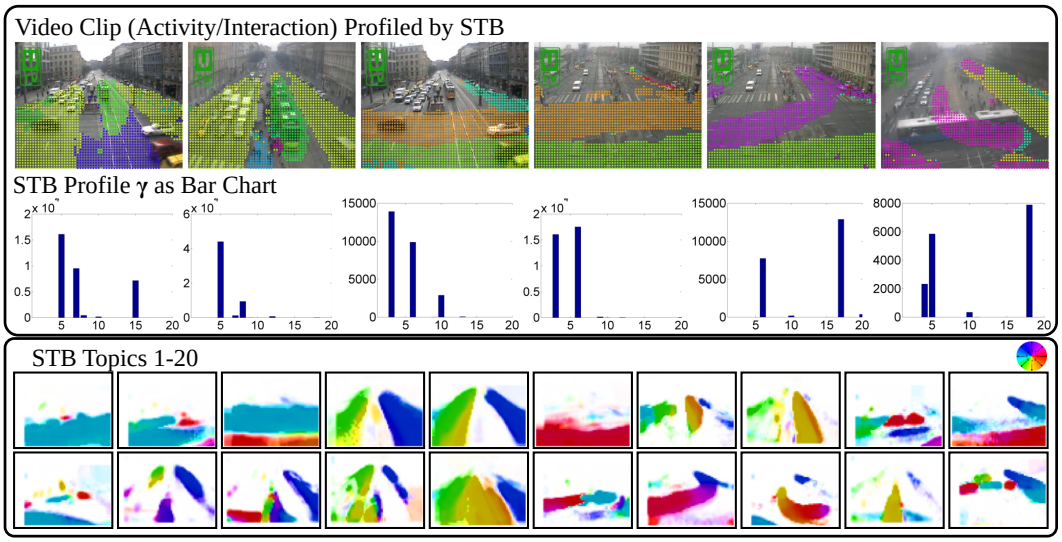


Figure 4.5: An illustration of behaviour profiling on STB. In the top block, visual words are profiled by STB and plotted as coloured dots. Notice that colors here indicate visual words belonging to individual actions in STB instead of motion direction. Profiling γ is also given as bar chart where x axis indexes STB actions. The bottom block illustrates the STB actions where colour patches indicate distribution of motion vectors.

query-by-example is achieved by sorting all clips in the cluster according to distance to the query.

4.3.2 Cross-Scene Classification

Given an existing annotated database of scenes modelled with our multi-layer framework, classification in a new scene s^* can now be achieved *without further annotation*. First s^* is associated to a cluster c^* (Section 4.2.1). Although s^* has no annotation, this reveals a set of semantically corresponding existing scenes from which annotation can meaningfully be borrowed. Classification can thus be achieved by any classifier, using all other scenes/clips and labels from cluster c^* as the labeled training set.

It should be noted that our cross-scene classification differs from Zheng and Jiang [48] and Zheng et al.[180] in: (1) we train on a **set** of source scenes before testing on a held-out scene rather than one source to one test scene. The conventional 1-1 approach requires implicitly the source and target scene to be *relevant* which must be manually identified. Our model is able to group relevant scenes automatically without requiring the user to know this as *a priori*; and (2) our model works in a transductive [41] manner. That is, it looks at target scene data during scene clustering, but without looking at the target data label. This weak assumption is more desirable in

practice because surveillance video data is often easy to collect but without any labelling, whilst the effort required for labelling is the bottleneck.

4.4 Multi-Scene Summarisation

In this section we present a multi-scene video summarisation algorithm that exploits the structure learned in Section 4.2 to compress cross-scene redundancy. All clips are represented by their profiles on STB. The general objective of multi-scene summarisation is to generate a *video skim* with at least one example of each distinct behaviour in the shortest possible summary. We generate independent summaries for each scene cluster (since different scene clusters are semantically dissimilar), and multi-scene summaries within each cluster (since scenes within a cluster are semantically similar).

4.4.1 K-Centre Summaries

The multi-scene summary video is of configurable length N_{sum}. Longer videos will show more distinct behaviours or more within-class variability of each behaviour. We compose the summary Σ of N_{sum} clips $\{\boldsymbol{\gamma}_j^{stb}\}, j \in \Sigma$ drawn from all scenes in the cluster. The objective is that all clips in the cluster $\{\boldsymbol{\gamma}_{js}^{stb}\}_{j,s\in\mathcal{C}}$ should be near to at least one clip in the summary (i.e., the summary is representative). Formally, this objective is to find the summary set Σ that minimizes the cost defined in Eq. (4.9) where $\mathcal{D}_\gamma$ is the L2 distance:

$$\mathcal{L} = \max_{j,s\in\mathcal{C}} \left(\max_{j'\in\Sigma} \quad \mathcal{D}_\gamma \left(\boldsymbol{\gamma}_{j'}^{stb}, \boldsymbol{\gamma}_{js}^{stb} \right) \right) \tag{4.9}$$

This is essentially a k-centre problem [181]. Since it is intractable to enumerate all combinations/potential summaries Σ, we adopt the 2-approximation algorithm [182] to this optimization. The resulting N_{sum} centres identify the summary clips.

4.5 Experiments

4.5.1 Datasets and Settings

Dataset

We collected 25 real traffic surveillance videos from publicly accessible online web-cameras in urban environment. These videos are combined with two surveillance video datasets Junction and Roundabout [8] for a total of 27 videos. Sample frames for each scene are illustrated in Figure

4.6(a). We trim each video to 18 000 frames in 10fps, of which 9 000 are used to learn the model and the remaining 9 000 frames are used for testing (query, classification and summarisation). For action learning we segment each training video into 25 frame clips, so 360 clips are generated for each scene. For both query and summarisation applications, we segment test videos into clips with 80 frames, so 112 clips for query and summarisation are generated from each scene. Thus, we have three types of video clips: (1) clips for unsupervised training of LDA; (2) clips for training cross-scene classification, retrieval and multi-scene summarisation, (Semantic Training Clips); and (3) clips for testing cross-the same tasks (Semantic Testing Clips). LDA clips are shorter (25 frames) to facilitate learning more cleanly segmented actions. Semantic clips are longer (80 frames) as a more human-scale user-friendly unit for visualisation and annotation.

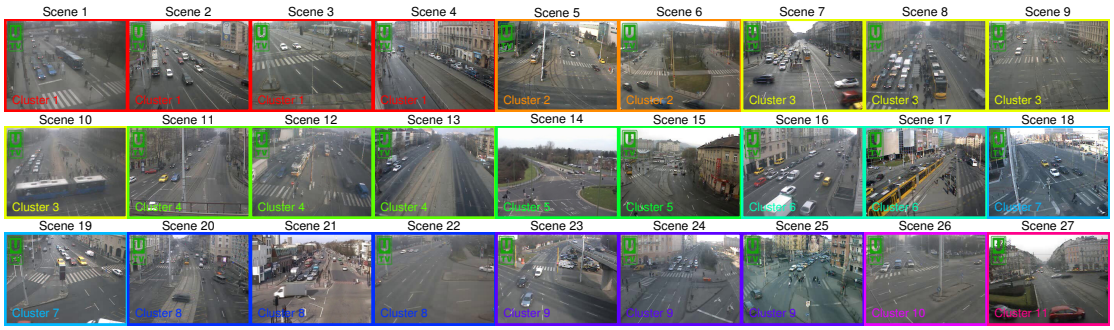


Figure 4.6: Example frames for our multi-surveillance video dataset with each scene assigned a reference number on top of the frame. The colour of bounding box and text in the bottom left indicates assigned cluster.

Learning Actions

We computed optical flow [83] for all videos by quantizing the scenes with 5×5 pixel cells and 8 directions. Local actions are learned from each video independently using LDA with $K = 15$ actions per scene.

Activity Annotation

Activity is a clip-level semantic concept defining the overall scene-action. Due to the semantic gap between activities in the video clip and (potentially task dependent) human interpretation, it is difficult to give video a concise and consistent semantic label (in contrast to human action [183] and event [40] recognition). Instead of annotating each video clip explicitly, we give a set of binary action tags (each representing the action of some objects within the scene) to each video clip as shown in Table 4.1. All the tags associated with vehicles have a sparse or dense

option. When there are less than three vehicles travelling in a clip, it is labelled as sparse, otherwise dense. Each unique combination of actions that exists in the labelled clips then defines a unique scene-level behaviour category. We explore this through multiple sets of annotations: an original annotation with 19 distinct tags, and subsequent coarser label sets derived by merge scheme 1 with 13 distinct tags and merge scheme 2 with 10 distinct tags. The action tags are given in Table 4.1. We exhaustively annotate video clips in two example scene clusters (3 and 7 as shown in Figure 4.6). Across the two clusters, there are 6 scenes with 112 clips per scene annotated (672 clips in total). In the original annotation case, there are 111 total behaviours identified. The distribution of behaviours are illustrated in Figure 4.7(a). However this number is more than necessary in terms of limited distinctiveness of the numerous entailed behaviours. By merging some action annotations we generate 59 or 31 (Merge Scheme 1 or 2 in Table 4.1) unique behaviours. It should be noted that the frequency of behaviours is rather imbalanced, as indicated by all the subfigures of Figure 4.7. There is also very limited overlap of behaviours between scene clusters 3 and 7. To assess annotation consistency and bias, we invited eight independent annotators to annotate all the video clips separately. We observe that the additional annotations are fairly consistent with the original annotation: with more than 80% agreement (Hamming distance) between the additional and the original annotations. Detailed analysis of these additional annotations are given in the supplementary material.

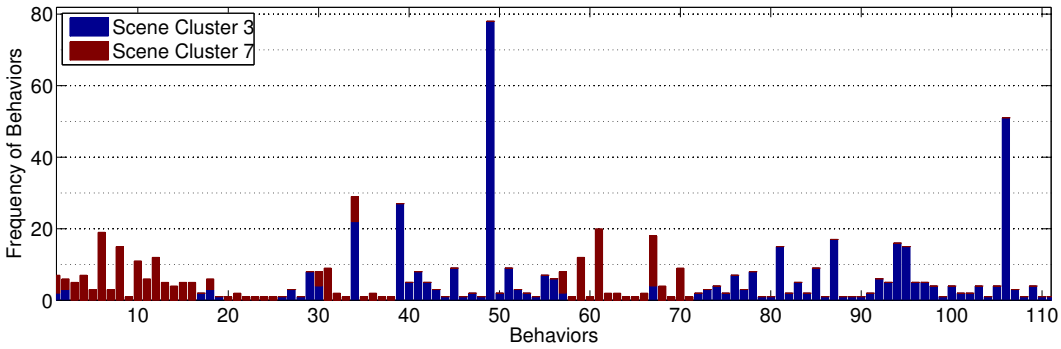
4.5.2 Multi-Layer Scene Clustering

Scene Level Clustering

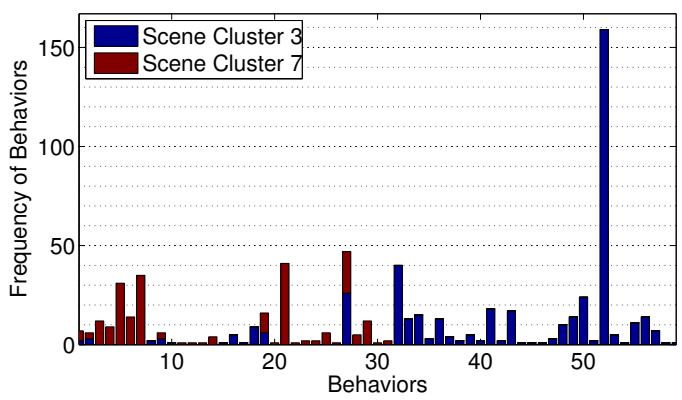
We first group the scenes into semantically similar clusters by spectral clustering. The similarity measurement between scenes is the number of corresponding actions, as defined in Section 4.2.1. The self-tuning spectral clustering automatically determines the appropriate number of clusters which, in the case of our 27-scene dataset, is 11 clusters. Figure 4.6 shows the results, in which semantically similar scenes are indeed grouped, e.g. Camera towards one direction at road junctions in Cluster 3, and unique views are separated into their own cluster, e.g. Cluster 11.

Learning A Shared Action Topic Representation

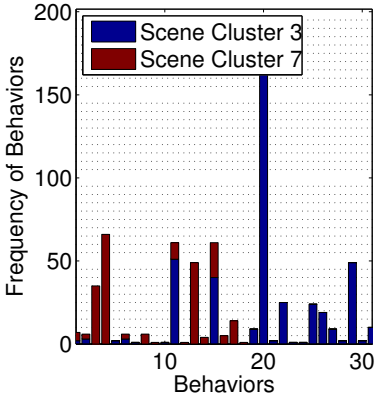
Within each scene cluster we unify the representation by computing a *shared action topic basis*. We automatically set the number of shared actions K^{stb} in each scene cluster with N_s scenes as



(a) Behaviour frequency: original annotation



(b) Behaviour frequency: merge scheme 1



(c) Behaviour frequency: merge scheme 2

Figure 4.7: Frequencies of behaviours of each category. (a), (b) and (c) illustrate the frequency of behaviours when varying the labelling criteria.

Table 4.1: Original annotation ontology and two merging schemes give multiple granularities of annotation.

No.	Original Annotation	Merge Scheme 1	Merge Scheme 2
1	Vehicle Left Sparse	Vehicle Left	Vehicle Horizontal
2	Vehicle Left Dense		
3	Vehicle Right Sparse	Vehicle Right	
4	Vehicle Right Dense		
5	Vehicle Up Sparse	Vehicle Up	Vehicle Vertical
6	Vehicle Up Dense		
7	Vehicle Down Sparse	Vehicle Down	
8	Vehicle Down Dense		
9	Vehicle Southeast Sparse	Vehicle Southeast	Vehicle SE& NW
10	Vehicle Southeast Dense		
11	Vehicle Northwest Sparse	Vehicle Northwest	
12	Vehicle Northwest Dense		
13	Vehicle Up2Right Turn	Vehicle Up2Right Turn	Vehicle Up2Right Turn
14	Vehicle Left2Up Turn	Vehicle Left2Up Turn	Vehicle Left2Up Turn
15	Vehicle Up2Left Turn	Vehicle Up2Left Turn	Vehicle Up2Left Turn
16	Tram Up	Tram Up	Tram Up
17	Tram Down	Tram Down	Tram Down
18	Pedestrian Horizontal	Pedestrian Horizontal	Pedestrian Horizontal
19	Pedestrian Vertical	Pedestrian Vertical	Pedestrian Vertical

$K^{stb} = coeff \times N_s$ where $coeff$ is set to 5. The discovered basis from an example cluster (Scene Cluster 3 shown in Figure4.6) with 4 scene members is illustrated in Figure 4.8. This figure reveals both actions unique to each scene (Topics 1-15) and actions common among multiple scenes (Topic 16-20). Thus some shared action topics are composed of single local/original topics, and others of multiple local topics.

4.5.3 Cross-Scene Query by Example and Classification

In this section we evaluate the ability of our framework to support two tasks: cross-scene query by example; and cross-scene activity classification. We compare our **Scene Cluster Model** (SCM) with a baseline **Flat Model** (FM). Our **Scene Cluster Model** first group scenes into scene clusters according to their relatedness and learns STB for every scene cluster. Video clips in each

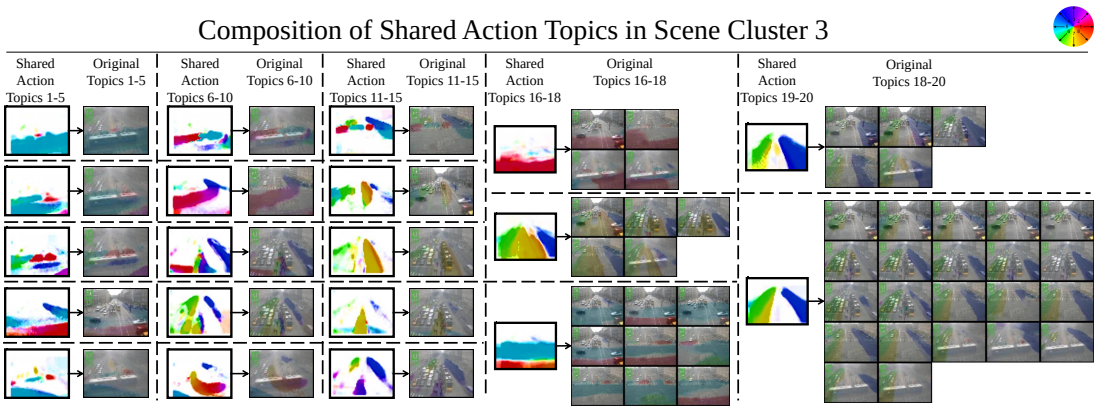


Figure 4.8: Example STB learned from Scene Cluster 3. Shared action topics may be composed of one or more local/original topics. Original topics are overlaid on background frame. Colour patches indicate distribution of motion vectors for a single action.

scene cluster are thus represented as topic profiles on the STB of the scene cluster. As with our model, a **Flat Model** first learns a local topic model per scene, however it then learns a single STB from all labelled scenes (6 scenes from 2 clusters) without scene level clustering, instead of one STB per-cluster. The only difference between SCM and FM is the absence of scene-level clustering in FM. Note that the Flat Model is a special case of our Scene Cluster Model with a single scene-level cluster. Moreover, the individual scenes are also a special case of our Scene Cluster Model with one scene per cluster.

Query by Example Evaluation

To quantitatively evaluate query by example, we exhaustively take each scene and each clip in turn as the query, and all other scenes are considered as the pool. All clips in the pool are ranked according to similarity (L2 distance on STB profile) to the query. Performance is evaluated according to how many clips with the same behaviour as the query clip are in the top N_t responses. We retrieve the best $N_t = 1 \cdots 200$ clips and calculate the *Average Precision* of each category for each N_t. MAP is computed by taking the mean value of *Average Precision* over all categories. The MAP curve by the top T responses to a query for both **Scene Cluster Model** (SCM) and **Flat Model** (FM) and Merge Scheme 1 and 2 are plotted in Figure 4.9. It is evident that for both Merge Scheme 1 and 2, the proposed scene cluster model (SCM) performs consistently better than the Flat Model (FM) regardless of number of top retrievals T. This is because in the Scene Cluster Model, the STB learned from this set of scenes are highly relevant to each scene in the cluster. In contrast, the Flat Model learns a single STB for all scenes making the STB less

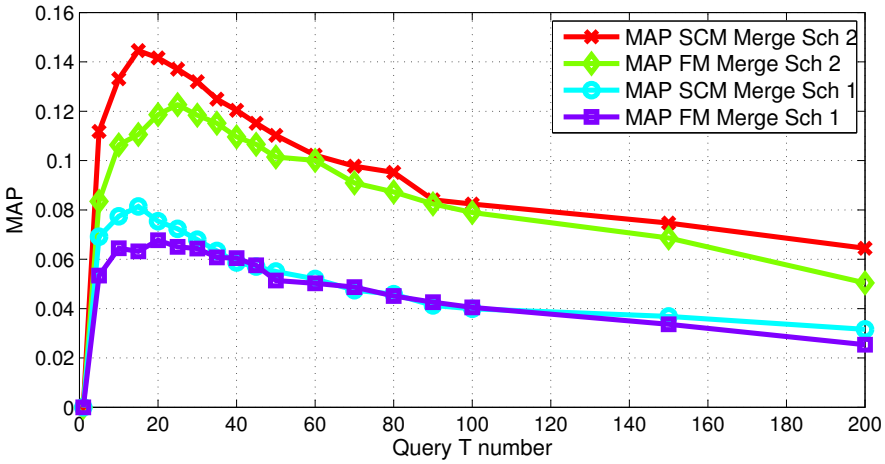


Figure 4.9: Query by example MAP with different number of retrievals.

relevant to each individual scene, hence less informative as a representation for retrieval.

Qualitative results are also given in Figure 4.10 by presenting 6 randomly chosen queries and their retrieved clips. Different types of behaviours are covered by query clips and most retrieved clips are semantically similar to query clips. The only exception is in the 3rd row where the query clip indicates traffic going east and turning from left to up. This is because there is no corresponding behaviour in the other scenes.

Classification Evaluation

In this experiment we quantitatively evaluate classification performance where the test scene has *no labels*. Successful classification thus depends correctly finding semantically related scenes and appropriately transferring labels from them (Section 4.3). We perform leave one scene out evaluation by holding out one scene as the unlabelled testing set, and predicting the labels for the test set clips using the labels in remaining scenes using the KNN classifier. The KNN K parameter is determined by cross validating among the remaining scenes. Classification performance is evaluated by the accuracy for each category of behaviour, averaged over all held out scenes.

From Table 4.2 we observe that at either granularity of annotation (59 or 31 categories), our **Scene Cluster Model** outperforms the **Flat Model** on average. This shows that again in order to borrow labels from other scenes for cross-scene classification, it is important to select relevant sources, which we achieve via scene clustering. The **Flat Model** is easily confused by the wider variety of scenes to borrow labels from, while our **Scene Cluster Model** structures similar scenes and borrows labels from only semantic related scenes to avoid 'negative transfer' [41].

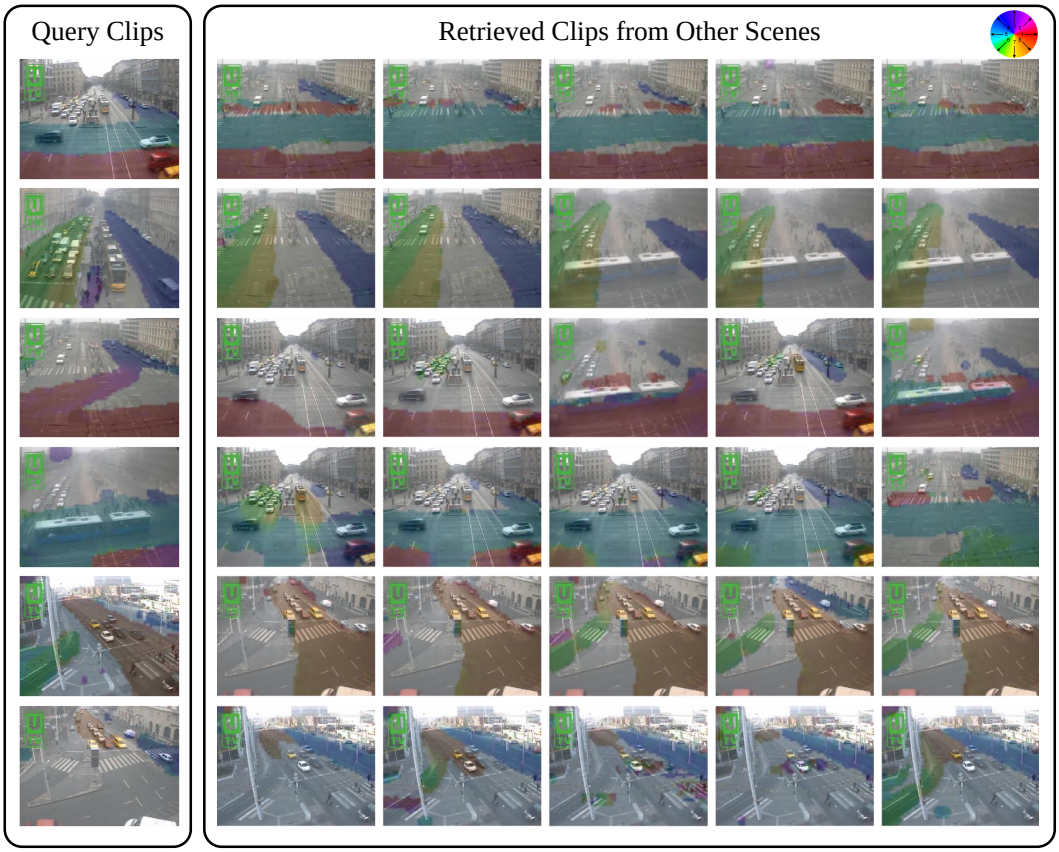


Figure 4.10: Examples of cross-scene query by example. The first column gives 6 query clips randomly chosen from 6 scenes. The right image matrix illustrates the retrieved clips from the remaining 5 scenes, sorted by distance to query from left to right in the matrix. Colour patches overlaid on the background indicates the visual words accumulated within a video clip.

Table 4.2: Cross-scene classification accuracy with 31 and 59 categories for both Scene Cluster Model (SCM) and Flat Model (FM).

Category	31		59	
	SCM	FM	SCM	FM
Scene 1	**55.36%**	50.89%	**42.86%**	40.18%
Scene 2	27.68%	**39.29%**	**18.75%**	16.96%
Scene 3	**49.11%**	41.96%	**39.29%**	37.50%
Scene 4	**54.46%**	46.43%	**37.50%**	36.61%
Scene 5	**30.36%**	26.79%	**17.86%**	**17.86%**
Scene 6	**38.39%**	25.00%	**20.54%**	12.50%
Average	**42.56%**	38.39%	**29.47%**	26.94%

4.5.4 Multi-Scene Summarisation

In the final experiment, we evaluate our multi-scene summarisation model against a variety of alternatives. We consider two conditions: In the first, we consider multi-scene summarisation within a scene cluster (Condition WC); in the second we consider unconstrained multi-scene summarisation including videos spanning multiple scene clusters (Condition AC).

Condition: Within-Cluster Summarisation (WC)

In this experiment we focus on the comparison between **Multi-Scene Model** and **Single-Scene Model** given various summarisation algorithms. The **Multi-Scene Model** represents all video clips from different scenes within a cluster with a single STB learned from the scene cluster while the **Single-Scene Model** represents each video with scene specific actions and the overall summary is the mere concatenation of summaries from each scene. Specifically, we compare the summarisation methods listed in Table 4.3.

Table 4.3: Summarisation schemes for **Condition WC**

Summarisation Method	**Description**
Random	This lower-bound picks clips randomly from multiple scenes to compose the summary
Single-Scene Graph	The overall summary is a concatenation of independent summaries for each video by doing recursive Normalised cut [184] on a graph constructed by taking each video clip as vertices and L2 distance between topic profile γ of each clip as edges. Here each video clip is represented by scene-specific local topics. This corresponds to Ngo et al.[185], but without temporal graph.
Single-Scene Kcenter	Similar to *Single-Scene Graph* method, but using Kcenter algorithm in Eq. (4.9) for summarisation instead of Normalised Cut.
Multi-Scene Graph	This model learns a STB to represent video clips from all scenes with STB profile. Then Normalised Cut is applied to cluster clips and find multi-scene summaries.
Multi-Scene Kcenter	Our full model builds a STB from all scenes within a cluster, then uses the Kcenter algorithm to select summary clips from all scenes.

Condition: Across-cluster Summarisation (AC)

In this experiment, analogous to query and classification, we focus on the comparison between **Flat Model** and **Scene Cluster Model** given different summarisation algorithms. The **Flat**

Model Learns a single STB from all scenes available without discrimination while **Scene Cluster Model** learns a STB per scene cluster. Specifically, we compare the summarisation schemes in Table 4.4.

Table 4.4: Summarisation schemes for **Condition AC**.

Summarisation Method	Description
Random	This picks clips randomly from multiple scenes to compose the summary
Flat Multi-Scene User Attention	Leverages the magnitude, spatial and temporal phase of optical flow vectors to index videos. This is the visual attention measurement of Ma et al.[186], Eq. (6). We tested the model on a combined video by concatenating each individual video.
Flat Multi-Scene Graph	This model uses Normalised Cut [184] to cluster all video clips represented as single STB profiles. This is similar to Ngo et al.[185].
Flat Multi-Scene Kcenter	Same as *Flat Multi-Scene Graph*, but using Kcenter to select summary clips.
Scene Cluster Multi-Scene Kcenter	Our full model clusters the scenes, learns STBs on each scene cluster, followed by Kcenter to summaries within each scene cluster

Settings

To systematically evaluate summarisation performance, we vary the length of the requested summary. In **Condition WC** the summary varies from 8 to 120 clips (64seconds to 16mins) out of overall 448 video clips (59.7mins) in Scene Cluster 3 (as shown in Figure 4.6(a)) and 224 video clips (29.9mins) in Scene Cluster 7. In **Condition AC** the summary varies from 6 to 120 clips (48seconds to 16mins) out of 672 video clips (89.7mins) total which is a combination of Scene Cluster 3 and 7. All video clips for summarisation are represented as topic profile γ. Recall that each local scene is learned with $K = 15$ topics and scene clusters with N_s scenes are learned with $K = coeff \times N_s$ topics where $coeff$ is set to 5 here. For fair comparison, flat model baselines are learned with the sum of the number of topics for each cluster.

Summarisation Evaluation

The performance is evaluated by the coverage of identified behaviours in the summary, averaged over 50 independent runs. Figure 4.11(a) and (b) show the results for multi-scene summarisation within two example clusters (**Condition WC**). Clearly our Multi-Scene Kcenter algorithm (red) outperforms the baselines: both Graph Method alternative (purple), and single-scene alter-

natives (dashed line). The performance margin is greater between multi-scene and single-scene models for the first cluster because there are four scenes here, so greater opportunity to exploit inter-scene redundancy. This validates the effectiveness of jointly exploiting multiple-scenes for summarisation. Figure 4.11(c) shows the result for multi-scene summarisation across both clusters (**Condition AC**): our **Scene Cluster Model** builds one summary for each cluster to exploit the expected greater volume of within-cluster redundancy. In contrast, the **Flat Model** builds one single summary, but for a much more diverse group of data, and the single-scene models have no across-cluster redundancy to exploit. Even in the flat case, our Kcenter model (in green) still outperforms all other alternatives (purple and magenta). It is also worth noting that the user attention model degenerates severely on our dataset due to the inability to extract semantic meaning from videos where pure motion strength is not informative enough to distinguish semantic behaviours.

4.5.5 Further Analysis

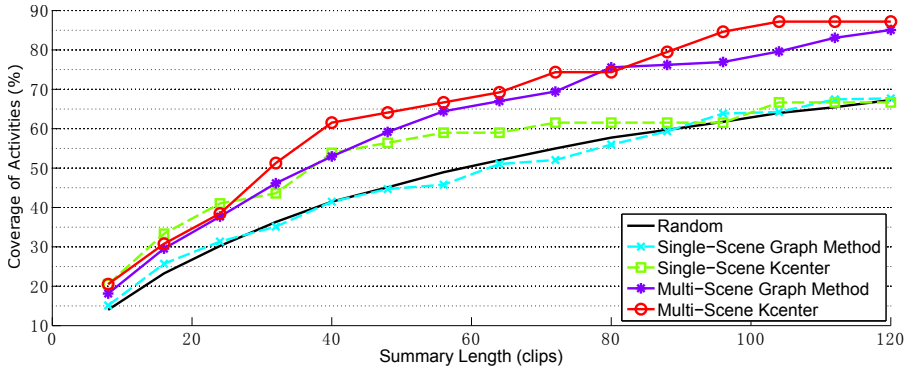
In this section, we further analyse the robustness of our framework, by varying key parameters, and investigate their impact on model performance.

Generalised Scene Alignment

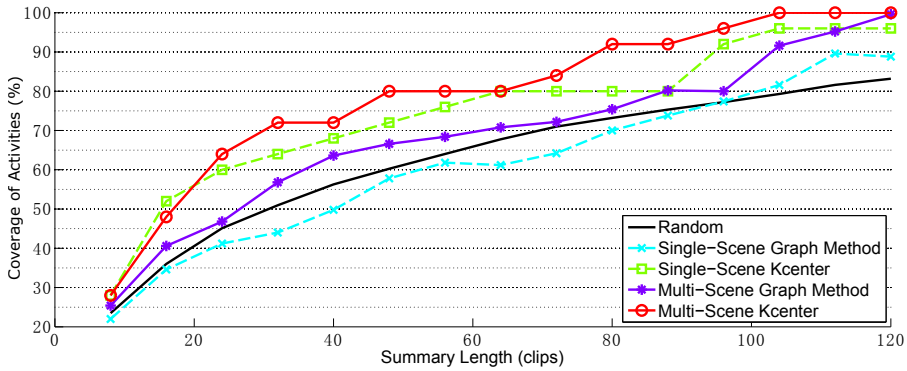
We assume currently that cameras are installed upright and only scaling and translational transform are applied to scene alignment. However, under more generally, rotational transforms may also be considered. To that end, one can consider a generalised scene alignment that includes a rotational parameter ϕ in the transformation. Recall that in section 4.2.1, we estimate the size of transformed topics. We can extend that to $N'_a = N_a \times h_s \times cos(\phi)$ and $N'_b = N_b \times h_s \times sin(\phi)$. The generalised transform matrix $\mathbf{H}$ is then defined as:

$$\mathbf{H} = \begin{bmatrix} h_s \cdot cos(\phi) & -h_s \cdot sin(\phi) & h_x \\ h_s \cdot sin(\phi) & h_s \cdot cos(\phi) & h_y \\ 0 & 0 & 1 \end{bmatrix} \tag{4.10}$$

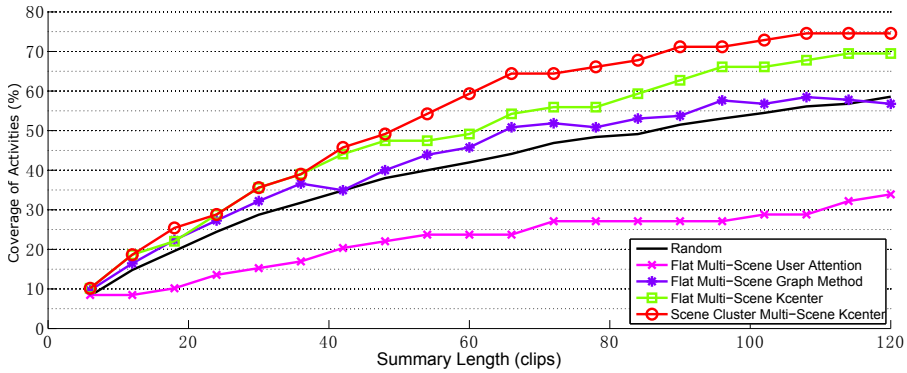
The procedure to transform a topic under this generalised alignment differs from the original alignment only in the estimation of direction d. To determine d given d', we represent quantized optical flow as vector $vec' = [cos(2\pi d'/N_m),\ sin(2\pi d'/N_m)]^\top$. Then we estimate the original flow vector $vec = \mathbf{H}^{*-1} vec'$ where $\mathbf{H}^*$ is a 2×2 matrix from the first two dimensions of $\mathbf{H}$ because translation does not change motion direction. We determine d by nearest neighbour as follows:



(a) Condition WC: Scene Cluster 3 (4 scenes total)



(b) Condition WC: Scene Cluster 7 (2 scenes total)



(c) Condition AC: All Scenes (6 scenes total)

Figure 4.11: Video summarisation results: Coverage of behaviours versus summary clip length.

$$\hat{d} = \underset{d=1\cdots N_m}{argmin} \left\| vec - \begin{bmatrix} cos(2\pi d/N_m) \\ sin(2\pi d/N_m) \end{bmatrix} \right\| \tag{4.11}$$

To align scene a to scene b with this generalised alignment, we can estimate parameters by maximizing the marginal likelihood of target document $\mathbf{X_b}$ given source topics β_a. Specifically, we denote the transform operation with specified parameters as $\mathbb{H}(\beta|h_s,h_x,h_y,\phi)$. Given target

document $\mathbf{X_b}$, the marginal likelihood is $p(\mathbf{X_b}|\alpha_a, \mathbb{H}(\beta_a|h_s,h_x,h_y,\phi))$ where α_a is the Dirichlet prior in scene a. Because scaling and translational parameters are computed by a closed-form solution (Eq. (4.3)), we only need to search $\hat{\phi} = \underset{\phi}{argmax}\ p(\mathbf{X_b}|\alpha_a, H(\beta_a|h_s,h_x,h_y,\phi))$. However, in our experiments with applying this generalised alignment process, we observed many local minima – suggesting that the rotational transform is under-constrained, and not very repeatable.

Scene Alignment Stability

We first evaluate the stability of scene-level alignment. Recall that given two scenes a and b, we firstly normalise each scene with geometrical transformation $\mathbf{H}^a_{norm}$ and $\mathbf{H}^b_{norm}$. The scene a to b transform is thus defined by:

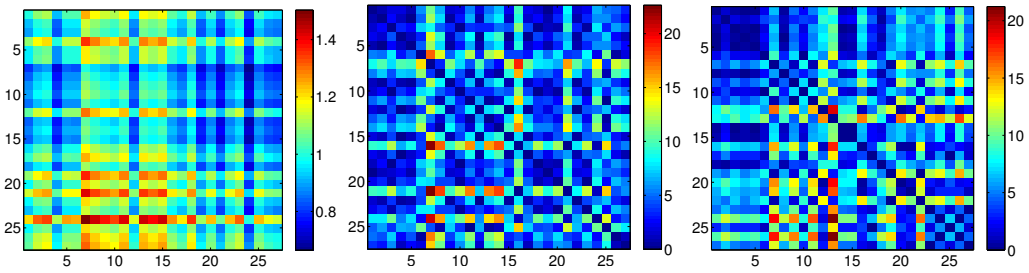
$$\mathbf{H}^{a2b} = \mathbf{H}^{b-1}_{norm} \cdot \mathbf{H}^a_{norm} = \begin{bmatrix} \frac{h^a_s}{h^b_s} & 0 & \frac{h^a_x}{h^b_s} - \frac{h^b_x}{h^b_s} \\ 0 & \frac{h^a_s}{h^b_s} & \frac{h^a_y}{h^b_s} - \frac{h^b_y}{h^b_s} \\ 0 & 0 & 1 \end{bmatrix} \tag{4.12}$$

We denote $h^{a2b}_s = \frac{h^a_s}{h^b_s}$, $h^{a2b}_x = \frac{h^a_x}{h^b_s} - \frac{h^b_x}{h^b_s}$, $h^{a2b}_y = \frac{h^a_y}{h^b_s} - \frac{h^b_y}{h^b_s}$. To evaluate the stability of this alignment, we randomly sample 50% of the original data from each scene and estimate again the parameters as $\tilde{h}^{a2b}_s$, $\tilde{h}^{a2b}_x$, $\tilde{h}^{a2b}_y$. We run this process for 20 times and calculate the Root Mean Square Error (RMSE), defined in Eq. (4.13) for h^{a2b}_s. RMSE for h_x and h_y are defined in the same way by replacing h^{a2b}_s with h^{a2b}_x and h^{a2b}_y respectively.

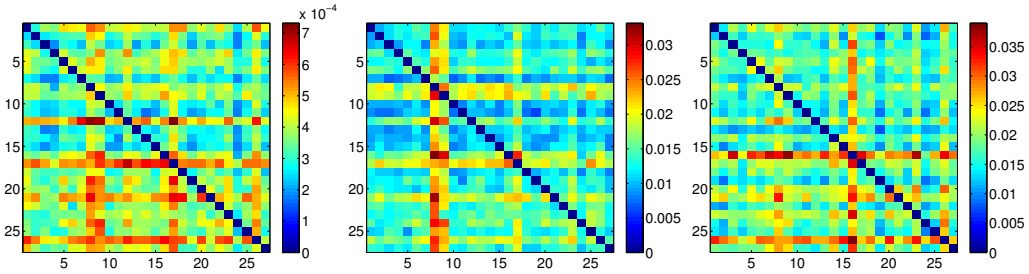
$$RMSE(h_s) = \sqrt{\frac{1}{20}\sum_{i=1}^{20}(\tilde{h}^{a2b}_{si} - h^{a2b}_s)^2} \tag{4.13}$$

We show both the absolute value of reference parameters and RMSE when aligning each pair of scenes in Figure4.12.

It is evident that most scene pairs are scaled between 0.7 and 1.5 (Figure 4.12(a)). The worst RMSE(h_s) among all scene pairs is 0.0007 (Figure 4.12(d)). The same observations can be made on variability of x translation and y translation with the largest RMSE(h_x) and RMSE(h_y) being 0.035 pixels or less while the absolute value of reference x and y translation are between 0 and 20 pixels. The small values of these deviations verify that the scene alignment model is robust and repeatable. Some examples of scene alignment are shown in Figure 4.13. Whilst the majority of activities are aligned well, some are less so. This is due to the limitation of a global rigid transform over a whole scene. Further extension could exploit individual activity centred alignment in addition to holistic scene alignment.



(a) Absolute reference value of scaling (b) Absolute reference value of x translation (c) Absolute reference value of y translation



(d) RMSE of scaling (e) RMSE of x translation (f) RMSE of y translation

Figure 4.12: Alignment and stability across all pairs of 27 scenes.

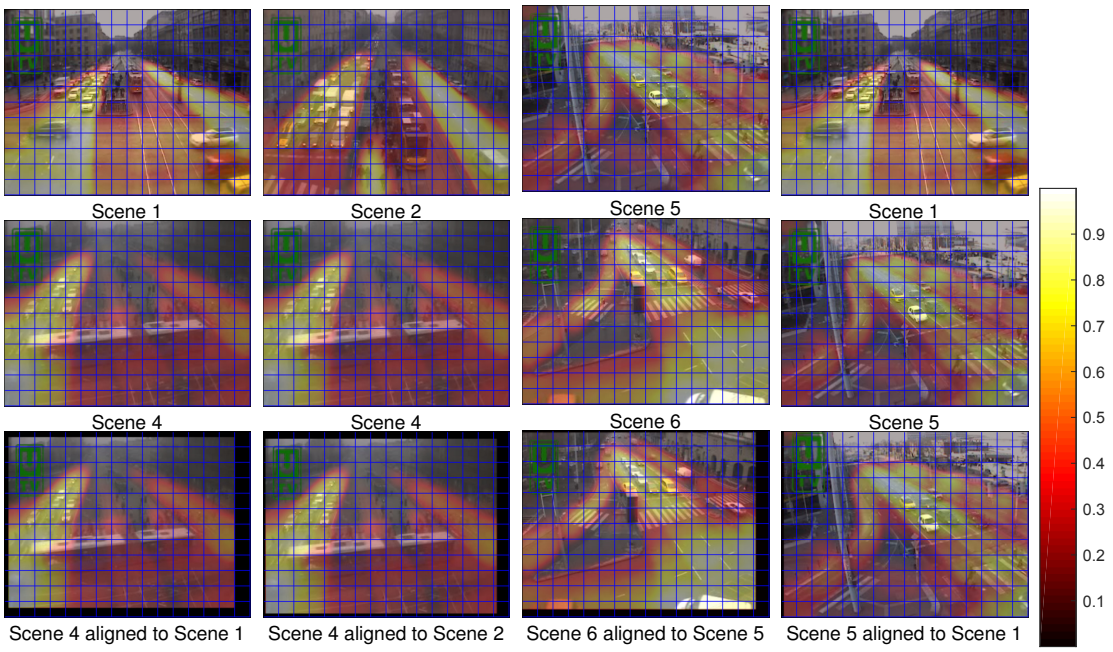
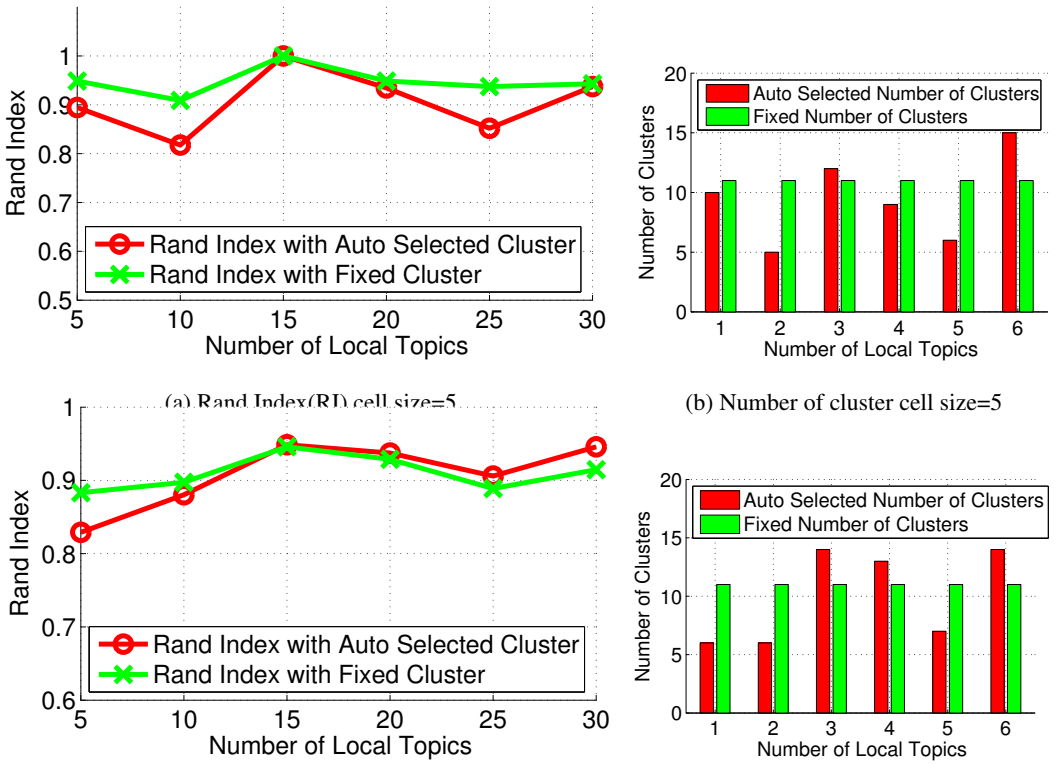


Figure 4.13: Examples of scene alignment pairs. Each column indicates one example alignment. The first row is the target scene, the second row is the source scene to be aligned/transformed and the last row is the source scene after alignment to the target. Both within scene cluster (first three columns, clusters 3, 3 and 7 respectively) and across cluster (fourth column, cluster 3 and 7) examples are presented. The overlaid heat map is the spatial frequency of visual words. A colorbar is attached to the right for reference however the number in the colorbar does not correspond to any physical measure.

Scene Cluster Stability

We tested the stability of scene-level clustering by varying cell size, number of local topics, and clustering strategy: (1) we compared visual word quantisation with 5×5 and 10×10 cell size; (2) we evaluated from 5 to 30 local topics in each scene by a step of 5; and (3) we performed self-tuning spectral clustering with two alternative settings. The first is that we allowed the model to automatically determine number of clusters and the second is that we fixed the number of clusters to the same as in the reference clustering, i.e. 15 local topics and 5×5 cell size. We measured the discrepancy between the results from automatic clustering and the reference clustering using the Rand Index [187]. It describes the discrepancy between two set partitions and is frequently used as the evaluation metric for clustering. The Rand Index is between 0 and 1, with the higher value indicating more similar between two partitions. If two partitions are exactly the same, the Rand Index is 1. We show the results on the stability test of scene-level clustering in Figure 4.14.



(a) Rand Index(RI) cell size=5

(b) Number of cluster cell size=5

(c) Rand Index(RI) cell size=10

(d) Number of cluster cell size=10

Figure 4.14: Stability of scene-level clustering.

For both cell size = 5 and = 10, automatic cluster selection generates consistent partitions (high Rand Index). So the framework is robust to motion quantisation cell size. However, it is

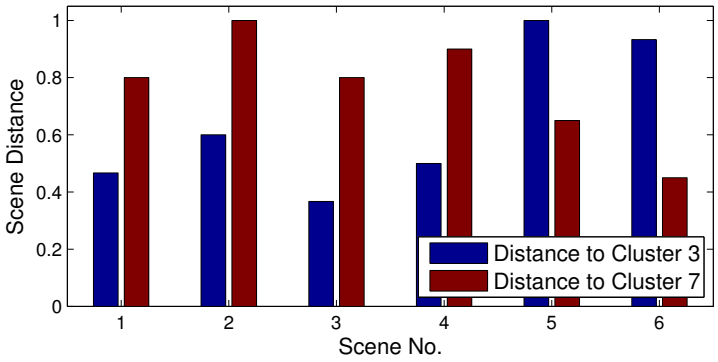


Figure 4.15: Association of held out-scenes to clusters. Scene 1-4 are held out from cluster 3, and scene 5-6 are held-out scenes from cluster 7. All held-out scenes are correctly associated.

also evident that automatic cluster number selection is less stable in determining the number of clusters as indicated by the red bars in Figure4.14(b) and (d). On the other hand, by fixing the number of clusters, the partitioning is more stable (consistent high Rand Index).

Associating New Scenes

Our model is able to group scenes according to the semantic relatedness if all the recorded data are available in advance. In addition, the model is capable of associating new scenes to existing clusters, e.g. given input from newly installed cameras at different locations, without the need to completely re-learn the model. This is achieved by comparing the local topics of a new scene to the STB in each scene cluster and choosing the cluster with highest relatedness. Only the updated cluster needs to be re-learned to incorporate the new scene. We tested this approach in Scene Clusters 3 and 7 by: (1) hold out each scene in turn as the candidate scene to be associated and learn STB in each cluster with the other scenes; (2) compute the relatedness between the held-out scene and both clusters using Eq. (4.6); and (3) associate the candidate scene to the cluster with the highest relatedness. We illustrate the result of this via the distance (defined as $1-$ relatedness) between held-out candidate scenes and clusters in Figure 4.15. It is evident that each held out scene is closer to its corresponding cluster, so 100% of scenes are associated correctly. However, this approach is limited to associating new scenes to existing scene clusters (scenes). A full online learning multi-scene model is desirable but also challenging and remains to be an open question.

STB Stability

Finally, we investigate the stability of learning the Shared Topic Basis (STB) with different number of shared topics. Recall that, in section 4.5.2, the number of STB topics for the Scene Cluster

Model (SCM) and the Flat Model (FM) is $coeff \times N_s$. Now let us change $coeff$ from 3 to 10 and evaluate how this affects the cross-scene classification accuracy for both annotation Scheme 1 (59 categories) and 2 (31 categories). The results are shown in Figure 4.16. It is evident that

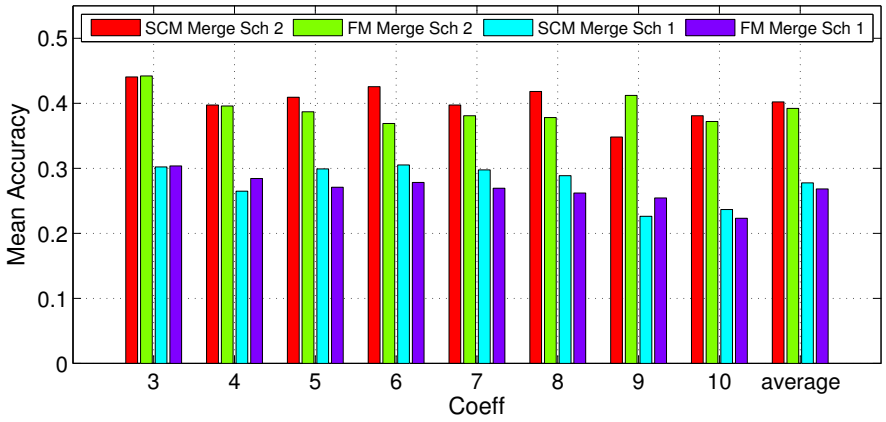


Figure 4.16: Effect of varying number of topics used. Classification accuracy of Scene Cluster Model (SCM) and Flat Model (FM).

for both 59 and 31 categories, our Scene Cluster Model is mostly better than Flat Model over a range of topic numbers.

4.6 Summary

In this chapter we introduced a framework for synergistically modelling multiple-scene datasets captured by multi-camera surveillance networks. It deals with variable and piece-wise inter-scene relatedness by semantically clustering scenes according to the correspondence of semantic activities; and selectively shares activities across scenes within clusters. Besides revealing the commonality and uniqueness of each scene, multi-scene profiling further enables typical surveillance tasks of query-by-example, behaviour classification and summarisation to be generalised to multiple scenes. Importantly, by discovering related scenes and shared activities, it is possible to achieve cross-scene query-by-example (in contrast to typical within-scene query), and to annotate behaviour in a novel scene without any labels – which is important for making deployment of surveillance systems scale in practice. Finally, we can provide video summarisation capabilities that uniquely exploit redundancy both within and across scenes by leveraging our multi-scene model.

There are still several limitations to this work. Firstly, in the current framework, scenes that can be grouped together are usually morphologically similar, which means the underlying motion

patterns and view angles are essentially similar. The semantic space defined in this chapter is thus unable to associate behaviours subject to more complex geometrical relationship, e.g. homography transformation, as attested in Section 4.5.5. Therefore, propogation of semantic labels in this framework is constrained by the geometrical layout. Secondly, in this work, motion information is mostly contributed by traffic. However studying human and crowd behaviour is becoming more interesting [15, 38] due to wide application in video content retrieval, human machine interaction, crime prevention and public security. However, compared with traffic, human and crowd behaviours are less regulated, coherent and independent of scene structures. Thus, exacting suitable features and discovering semantics to deal with the new situations are non-trivial tasks. To this end, the next chapter discusses analysing human behaviours. More importantly, with text induced semantic representation, we enable knowledge transfer between semantic categories to recognise novel classes without visual training data.

Chapter 5

Semantic Space for Zero-Shot Action Recognition

The preceding two chapters have demonstrated that discovering semantic space can benefit cross/multi-scene traffic behaviour analysis. However, both chapters focus on far field of view surveillance scenes where the behaviours are contributed mainly by the rigid movement of vehicles and pedestrians. The techniques studied in the previous chapters can not be trivially employed to help recognise human actions in a close-up view. Therefore, in this chapter, we address the human action recognition problem and more importantly study how semantic space discovered from text can help the action recognition.

Analysing human actions requires different techniques introduced for surveillance scenes due to the articulated movement of human body in conjunction with camera motion. Therefore, constructing discriminative feature is essential for good performance in human action recognition. More importantly, the number of categories for action recognition is growing rapidly and it has become increasingly hard to label sufficient training data for learning conventional models for all categories. Instead of collecting ever more data and labelling them exhaustively for all categories, an attractive alternative approach is ‘zero-shot learning (ZSL). Existing ZSL studies focus primarily on still images, and attribute-based semantic representations. In contrast, we explore word-vectors as the shared semantic space to embed videos and category labels for ZSL action recognition. This is a more challenging problem than existing ZSL of still images and/or attributes, because the mapping between video space-time features of actions and the semantic space is more complex and harder to learn for the purpose of generalising over any cross-category domain shift. To solve this generalisation problem in ZSL action recognition, we investigate a

series of synergistic improvements to the standard ZSL pipeline. First, we enhance significantly the semantic space mapping by proposing manifold-regularised regression and data augmentation strategies. Second, we evaluate two existing post processing strategies (transductive self-training and hubness correction), and show that they are complementary. We evaluate extensively our model on a wide range of human action datasets including HMDB51, UCF101, OlympicSports and event dataset including CCV and TRECVID MED 13. The results demonstrate that our approach achieves the state-of-the-art zero-shot action recognition performance with a simple and efficient pipeline, and without supervised annotation of attributes. Finally, we present in-depth analysis into why and when zero-shot works, including demonstrating the ability to predict cross-category transferability in advance.

The remainder of this chapter is organised as below: Section 5.1 introduces the procedure to learn visual-semantic model on labelled known categories. In Section 5.2 we introduce transductive algorithm to improve zero-shot prediction performance. Experiment on 4 popular action datasets and one event detection dataset are conducted in Section 5.3 with a summary given in Section 5.4.

5.1 Learning Visual-Semantic Model

Table 5.1: Basic notations for zero-shot action recognition.

Notation	Description
$\mathbf{X} \in \mathbb{R}^{d_x \times N}$; $\mathbf{x}_i$	Visual feature matrix for N instances; Column representing the i-th instance
$\mathbf{y} \in \mathbb{Z}^{1 \times N}$; y_i	Integer class labels for N instances; Scalar representing the i-th instance
$\mathbf{Z} \in \mathbb{R}^{d_z \times N}$; $\mathbf{z}_i$	Semantic embedding for N instances; Column representing the i-th instance
$\mathbf{K} \in \mathbb{R}^{N \times N}$	Kernel matrix
$\mathbf{A} \in \mathbb{R}^{d_z \times N}$	Regression coefficient matrix
$f : \mathbf{X} \rightarrow \mathbf{Z}$	visual-to-semantic mapping function
$g : \mathbf{y} \rightarrow \mathbf{Z}$	Class name embedding function
$\lambda_A \in \mathbb{R}$	Ridge regression regularizor
$\lambda_I \in \mathbb{R}$	Manifold regression regularizor
$N_K^G \in \mathbb{Z}^+$	KNN Graph parameter for manifold regularizor
$N_K^{st} \in \mathbb{Z}^+$	KNN parameter for Self-Training procedure

First of all, we give an overview of out zero-shot action recognition framework in Figure 5.1 and define the frequently used notations in Table 5.1. We have labelled training data in auxiliary

dataset $\mathbf{X}_{tr}^{aux}$, additional labelled augment dataset $\mathbf{X}_{1}^{aug}$ and unlabelled testing data in target dataset $\mathbf{X}_{te}^{trg}$. Each of the labelled and unlabelled data is accompanied with a class index, e.g. 'brush hair', 'biking'. We denoted the class indices as $\mathbf{y}_{tr}^{aux}$, $\mathbf{y}_{1}^{aug}$ and $\mathbf{y}_{te}^{trg}$ respectively. We assume the access to the labelled data $\mathbf{X}_{tr}^{aux}$ and $\mathbf{X}_{1}^{aug}$ with corresponding labels $\mathbf{y}_{tr}^{aux}$, $\mathbf{y}_{1}^{aug}$ in the training phase. The objective is to use all labelled information to help classify unlabelled data $\mathbf{X}_{te}^{trg}$ into a set of pre-defined categories $\mathbf{y}_{te}^{trg}$ (aka unknown classes). Importantly, the unlabelled classes are disjoint from any seen data at training time: $\mathbf{y}^{tr} \cap \mathbf{y}^{te} = \emptyset$. This learning strategy is often referred to as zero-shot learning (ZSL). In addition to the standard definition of zero-shot learning, we make a further assumption that we have the access to unlabelled testing data $\mathbf{X}_{te}^{trg}$ but not the labels $\mathbf{y}_{te}^{trg}$ in the training phase which is named as 'transductive' setting. Specifically, in the training phase I, auxiliary labelled data $\mathbf{X}_{tr}^{aux}$ is first augmented by data from augment dataset $\mathbf{X}_{1}^{aug}$ to form all labelled training dataset $\mathbf{X}^{tr}$ and $\mathbf{y}^{tr}$. We construct a K Nearest Neighbour (KNN) graph on all labelled and unlabelled data in visual feature space to model the underlying manifold structure. In the training phase II, we create prototypes for known classes which are class names embedded in semantic space via a an embedding function $\mathbf{Z}^{tr} = g(\mathbf{y}^{tr})$. Then we learn a visual-to-semantic mapping $f : \mathbf{X}^{tr} \rightarrow \mathbf{Z}^{tr}$ as manifold regularised regression. In the testing phase, prototypes for unknown classes are first generated by semantic embedding $g(\mathbf{y}^{te})$. Then target test/unlabelled data $\mathbf{X}_{te}^{trg}$ are projected into semantic space via $f(\mathbf{X}_{te}^{trg})$. Finally simple nearest neighbour (NN) classifier is adopted to categorise test data as the label of closest prototype. On top of NN classifier, self-training and hubness corrections are adopted at testing phase to further improve mitigate domain shift problem.

5.1.1 Semantic Embedding Space

To bridge the gap between disjoint training and testing classes, we establish a semantic embedding space $\mathbf{Z}$ based on word-vectors. In particular we use a neural network [24] trained on a 100 billion word corpus to realise a mapping $g : \mathbf{y} \rightarrow \mathbf{Z}$ that produces a unique d_z dimensional encoding vector of each dictionary word.

Compound Names

The above procedure only deals with class names that are unigram dictionary words. To process compound names commonly occurring in action datasets, e.g. 'brush hair' or 'ride horse', that do not exist as individual tokens in the corpus, we exploit compositionally of the semantic space

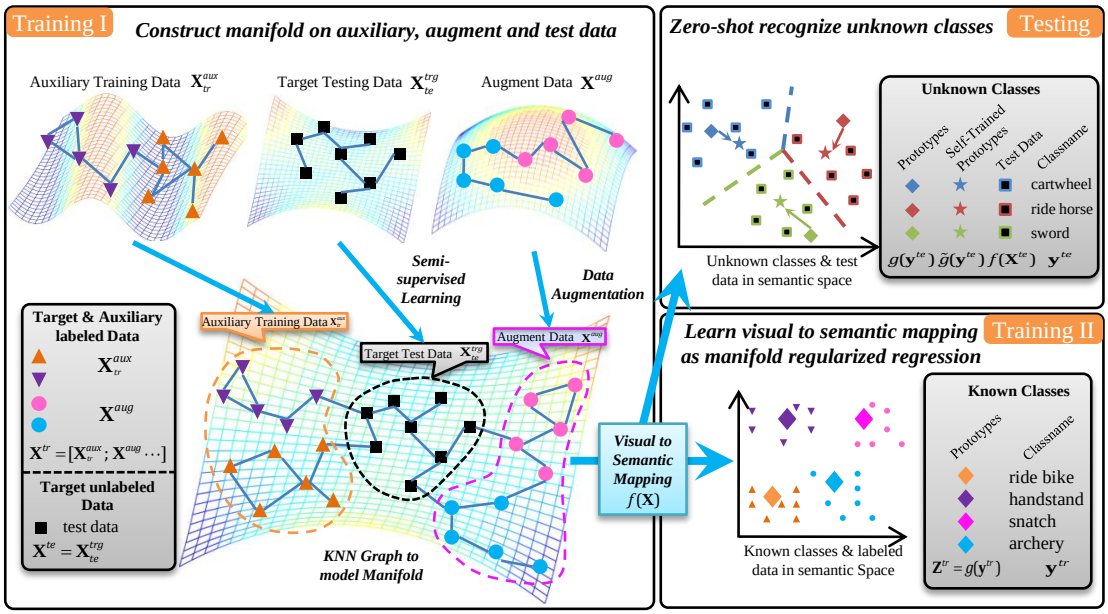


Figure 5.1: Zero-Shot Action Recognition Pipeline.

[188]. Various composition methods have been proposed [188, 189] including additive, multiplicative and others, but our experiments showed no significant to using others besides addition, so we stick with simple additive composition.

Suppose the ith class name y_i is composed of words $\{y_{ij}\}_{j=1\cdots w}$. We generate a single d_z dimensional vector $\mathbf{z}$ out of the word-vector y_i by summing the word-vectors for constituent words $\{y_{ij}\}$:

$$\mathbf{z}_i = \frac{1}{w} \cdot \sum_{j=1}^{w} g(y_{ij}) \tag{5.1}$$

5.1.2 Visual-to-Semantic Mapping

Mapping by Regression

In order to map video features into the semantic embedding space constructed above, we train a regression model $f : X \rightarrow Z$ from d_x dimensional low-level visual feature space to the d_z dimensional embedding space. The regression is trained using training instances $\mathbf{X}^{tr} = \{\mathbf{x}_i\}_{i=1\cdots n_l}$ and the corresponding embedding $\mathbf{Z}^{tr} = g(\mathbf{y}^{tr})$ of the instance class name y as the target value. Various methods have previously been used for this task including linear support vector regression (SVR) [139, 22] and more complex multi-layer neural networks [26, 141, 142]. Since we will use fisher vector encoding [92] for features $\mathbf{X}$, we can easily apply simple linear regression for $f(\cdot)$. Specifically, we use l_2 regularized linear regression (aka ridge regression) to learn the visual-to-semantic mapping.

Kernel Ridge Regression

The fisher vector encoding generates a very high dimensional feature $2 \times d_{descr} \times N_k$ where N_k is the number of components in the Gaussian Mixture Model (GMM) and d_{descr} is the dimension of raw descriptors. This usually results in many more feature dimensions than training samples. Thus we use the representer theorem [95] and formulate a kernelized ridge regression with a linear kernel in Eq (5.2). The benefit of kernelised regression is to reduce computation as the closed-form solution to $\mathbf{A}$ only involves computing the inverse of a $N \times N$ rather than a $d_x \times d_x$ matrix where $N < d_x$.

$$k(x_i, x_j) = \sum_{d=1}^{d_x} (x_{id} \cdot x_{jd}) \tag{5.2}$$

The visual features $\mathbf{x}$ can be then projected into semantic space via Eq (5.3) where $\mathbf{a}_j$ is the jth column of regression parameter matrix $\mathbf{A}$.

$$f(\mathbf{x}) = \sum_{j=1}^{n_l} \mathbf{a}_j k(\mathbf{x}, \mathbf{x}_j) \tag{5.3}$$

To improve the generalisation of the regressor, we add the l_2 regularizer $||f||^2_{\mathbf{K}} = Tr(\mathbf{AKA}^\top)$ to reduce overfitting by penalising extreme values in the regression matrix. This gives the kernel ridge regression loss:

$$\begin{aligned} &\min_f \frac{1}{n_l} \sum_{i=1}^{n_l} ||\mathbf{z}_i - f(\mathbf{x}_i)||_2^2 + \gamma ||f||_K^2 \\ &\min_{\mathbf{A}} \frac{1}{n_l} Tr\Big((\mathbf{Z} - \mathbf{AK})^\top (\mathbf{Z} - \mathbf{AK})\Big) + \gamma Tr(\mathbf{AKA}^\top) \end{aligned} \tag{5.4}$$

where the regression targets are generated by the vector representation of each class name $\mathbf{z}_i = g(y_i)$ and $\mathbf{Z} = [\mathbf{z}_1\, \mathbf{z}_2 \cdots]_{d_z \times n_l}$, $\mathbf{A}$ is the $d_z \times n_l$ regression coefficient matrix, $\mathbf{K}$ is the $n_l \times n_l$ kernel matrix and n_l is the number of labelled training instances. The loss function is convex with respect to the $\mathbf{A}$. Taking derivatives w.r.t $\mathbf{A}$ and setting the gradient to 0 leads to the following closed-form solution where $\mathbf{I}$ is the identity matrix.

$$\mathbf{A} = \mathbf{Z}\,(\mathbf{K} + \gamma_A n_l \mathbf{I})^{-1} \tag{5.5}$$

The above mapping by Kernel Ridge Regression provides a simple solution to embed visual instances into semantic space. However the simple ridge regression only considers limited labelled training data $\mathbf{X}^{tr}$ without exploiting the underline structure of the manifold on both

labelled and unlabelled data nor any additional related labelled data from other datasets. In the following sections, we introduce two approaches to improve the quality of mapping: (1) *Manifold-Regularized Regression*; and (2) *Data Augmentation*.

Manifold Regularized Regression

As discussed earlier, conventional regularization provides poor ZSL due to disjoint training and testing classes. To improve recognition of testing classes, we explore transductive semi-supervised regression. The idea is to exploit unlabelled testing data $\mathbf{X}^{te}$ to discover the manifold structure in the zero-shot classes, and preserve this structure in the semantic space after visual-semantic mapping. Therefore, this is also known as manifold regularization. Note that we use *labelled* to refer to training data $\mathbf{X}^{tr}$ and *unlabelled* to refer to testing data $\mathbf{X}^{te}$. So we use semi-supervised manifold regularization in a transductive way, requiring access to the unlabelled/testing data $\mathbf{X}^{te}$ during the training phase.

To that end, we introduce manifold laplacian regularization [57] into the ridge regression formulation. This additional regularization term ensures that if two videos are close to each other in the visual feature space, this relationship should be kept in the semantic space as well.

We model the manifold by constructing a symmetric K nearest neighbour (KNN) graph $\mathbf{W}$ on the all $n_l + n_u$ instances where $n_l = |\mathbf{T}^{tr}|$ denotes the number of labelled training instances and $n_u = |\mathbf{T}^{te}|$ denotes the number of unlabelled testing instances. The KNN Graph is constructed by first computing a linear kernel matrix between all instances. Then for each instance we select the top K neighbours and assign an edge between these nodes. This gives us a directed graph which is then symmetrized by converting to an undirected graph by connecting nodes with any directed edge between them. Let $\mathbf{D}$ be a diagonal matrix with $d_{ii} = \sum_{j=1}^{n_l+n_u} w_{ij}$, we get the graph laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{W}$. The manifold regularizer is then written as:

$$\begin{aligned} ||f||_I^2 &= \frac{1}{2}\sum_{i,j}^{n_l+n_u} w_{ij}||f(\mathbf{x}_i) - f(\mathbf{x}_j)||_2^2 \\ &= \frac{1}{2}\sum_{i,j} w_{ij} f^\top(\mathbf{x}_i) f(\mathbf{x}_i) + \frac{1}{2}\sum_{i,j} w_{ij} f^\top(\mathbf{x}_j) f(\mathbf{x}_j) \\ &\quad - \sum_{i,j} w_{ij} f^\top(\mathbf{x}_i) f(\mathbf{x}_j) \\ &= \sum_i d_{ii} f^\top(\mathbf{x}_i) f(\mathbf{x}_i) - \sum_{i,j} w_{ij} f^\top(\mathbf{x}_i) f(\mathbf{x}_j) \end{aligned} \tag{5.6}$$

Further denoting $\mathbf{f} = [f(\mathbf{x}_1)\ f(\mathbf{x}_2) \cdots\ f(x_{n_l+n_u})] = \mathbf{AK}$. Eq. (5.6) can be rewritten in matrix

form as:

$$\begin{aligned} ||f||_I^2 &= Tr(\mathbf{f}^\top \mathbf{f}\mathbf{D}) - Tr(\mathbf{f}^\top \mathbf{f}\mathbf{W}) \\ &= Tr(\mathbf{f}^\top \mathbf{f}\mathbf{L}) \\ &= Tr(\mathbf{K}^\top \mathbf{A}^\top \mathbf{A}\mathbf{K}\mathbf{L}) \end{aligned} \tag{5.7}$$

where $\mathbf{K}$ is a $(n_l + n_u) \times (n_l \times n_u)$ dimensional kernel matrix constructed upon all labelled and unlabelled instances via Eq (5.2). Combining all regularization terms we obtain the overall loss function in Eq (5.8), where for simplicity we denote $\mathbf{J} = \begin{bmatrix} \mathbf{I}_{n_l \times n_l} & \mathbf{0}_{n_l \times n_u} \\ \mathbf{0}_{n_u \times n_l} & \mathbf{0}_{n_u \times n_u} \end{bmatrix}$ and $\tilde{\mathbf{Z}} = [\mathbf{Z}^{tr} \quad \mathbf{0}_{d_z \times n_u}]$. The final loss function can be thus written in the matrix form as:

$$\begin{aligned} \min_{\mathbf{A}} \frac{1}{n_l} Tr\left((\tilde{\mathbf{Z}} - \mathbf{A}\mathbf{K}\mathbf{J})^\top (\tilde{\mathbf{Z}} - \mathbf{A}\mathbf{K}\mathbf{J})\right) + \gamma_A Tr(\mathbf{A}\mathbf{K}\mathbf{A}^\top) \\ + \frac{\gamma_I}{(n_l + n_u)^2} Tr(\mathbf{K}^\top \mathbf{A}^\top \mathbf{A}\mathbf{K}\mathbf{L}) \end{aligned} \tag{5.8}$$

The loss function is convex w.r.t. the $d_z \times (n_l + n_u)$ regression coefficient matrix $\mathbf{A}$. A closed-form solution to $\mathbf{A}$ can be obtained in the same way as Kernel Ridge Regression.

$$\mathbf{A} = \tilde{\mathbf{Z}} \left(\mathbf{K}\mathbf{J} + \gamma_A n_l \mathbf{I} + \frac{\gamma_I n_l}{(n_l + n_u)^2} \mathbf{K}\mathbf{L} \right)^{-1} \tag{5.9}$$

Eq (5.9) provides an efficient way to learn the visual-to-semantic mapping due to the closed-form solution compared to alternative iterative approaches [139, 23]. At testing time, the mapping can be efficiently applied to project new videos into the embedding with Eq. (5.3). Note when $\gamma_I = 0$ manifold regression becomes exactly kernel regression.

Improving the Embedding with Data Augmentation

As discussed, the mapping often generalises poorly because: (1) actions are visually complex and ambiguous; and (2) even a mapping well learned for training categories may not generalise well to testing categories as required by ZSL, because the volume of training data is small compared to the complexity of a general visual-to-semantic space mapping. The manifold regression described previously ameliorates the latter issues, but we next discuss a complementary strategy of data augmentation.

Another way to further mitigate both of these problems is by augmentation with any available additional dataset which need not contain classes in common with the target testing dataset.

This will provide more data to learn a better generalising regressor $\mathbf{z} = f(\mathbf{x})$. We formalise the data augmentation problem as follows. To recognise actions classes in HMDB51, there are n_{aux} additional augment datasets $\{\mathbf{X}_i^{aug}, \mathbf{y}_i^{aug}\}$, e.g. UCF101, Olympic Sports and CCV. We propose to improve the regression by merging auxiliary training data and all available augment sets. The augment dataset class names $\mathbf{y}_i^{aug}$ are projected into the embedding space with $\mathbf{Z}_i^{aug} = g(\mathbf{y}_i^{aug})$. The additional augment instances $\mathbf{X}^{aug}$ are aggregated with the auxiliary training data as $\mathbf{X}^{tr} = [\mathbf{X}_{tr}^{aux}\, \mathbf{X}_1^{aug} \cdots \mathbf{X}_n^{aug}]$ and $\mathbf{Z}^{tr} = [\mathbf{Z}_{tr}^{aux}\, \mathbf{Z}_1^{aug} \cdots \mathbf{Z}_n^{aug}]$ where $\mathbf{Z}_{tr}^{aux} = g(\mathbf{y}_{tr}^{aux})$. The augmented training data $\mathbf{X}^{tr}$ and class embeddings $\mathbf{Z}^{tr}$ are used together to train the regressor f. It is worth noting that we don't re-train the Fisher Vector codebook by augmenting additional dataset.

To formulate the loss function in matrix form we denote $n_l^{aux} = |\mathbf{y}_{tr}^{aux}|$, $n_u^{trg} = |\mathbf{y}_{te}^{trg}|$, $n_l^{aug} = \sum_i |\mathbf{y}_i^{aug}|$. Let $\tilde{\mathbf{K}}$ be the $(n_l^{aux} + n_u^{trg} + n_l^{aug}) \times (n_l^{aux} + n_u^{trg} + n_l^{aug})$ dimensional kernel matrix on all target and auxiliary data, and $\tilde{\mathbf{L}}$ is the corresponding graph laplacian. We then write the block structured $\tilde{\mathbf{J}}$ matrix as:

$$\tilde{\mathbf{J}} = \begin{bmatrix} \mathbf{I}_{(n_l^{aux}+n_l^{aug}) \times (n_l^{aux}+n_l^{aug})} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}_{n_u^{trg} \times n_u^{trg}} \end{bmatrix} \tag{5.10}$$

The loss function of manifold regularized regression with data augmentation is thus written in a matrix form as:

$$\begin{aligned} \min_{\mathbf{A}} \frac{1}{(n_l^{aux}+n_l^{aug})} Tr\left((\tilde{\mathbf{Z}}^{tr} - \mathbf{A}\tilde{\mathbf{K}}\tilde{\mathbf{J}})^\top (\tilde{\mathbf{Z}}^{tr} - \mathbf{A}\tilde{\mathbf{K}}\tilde{\mathbf{J}})\right) + \gamma_A Tr(\mathbf{A}\tilde{\mathbf{K}}\mathbf{A}^\top) \\ + \frac{\gamma_I}{(n_l^{aux}+n_{te}^{trg}+n_l^{aug})^2} Tr(\tilde{\mathbf{K}}^\top \mathbf{A}^\top \mathbf{A}\tilde{\mathbf{K}}\tilde{\mathbf{L}}) \end{aligned} \tag{5.11}$$

In the same way as before, we obtain the closed-form solution to $\mathbf{A}$:

$$\mathbf{A} = \tilde{\mathbf{Z}}^{tr}\left(\tilde{\mathbf{K}}\tilde{\mathbf{J}} + \gamma_A(n_l^{aux}+n_l^{aug})\mathbf{I} + \frac{\gamma_I(n_l^{aux}+n_l^{aux})}{(n_l^{aux}+n_u^{trg}+n_l^{aug})^2}\tilde{\mathbf{K}}\tilde{\mathbf{L}}\right)^{-1} \tag{5.12}$$

where by setting $\gamma_I = 0$ we obtain a kernel ridge regression with only data augmentation.

Relation to Multi-Task Learning

The multi-task learning (MTL) concept adopted by Simonyan and Zisserman [68] is very similar to our data augmentation as both aim to learn a more robust and generalisable model. However, we note there are several key aspects which differentiate the two strategies. Firstly, the MTL in Simonyan and Zisserman [68] defines two classification loss, i.e. two softmax loss and take the

sum as the final loss for training the network. This strategy does not differentiate the overlapped categories between two datasets. As an alternative strategy, Simonyan and Zisserman [68] proposed to manually prune out overlapped categories and merge two datasets which is not scalable to larger and more training datasets. In contrast, our model is able to automatically expand our training pool by including additional datasets in the semantic space.

5.2 Transductive Zero-Shot Prediction

Given the trained mappings $f(\cdot)$ and $g(\cdot)$ we can now complete the zero-shot learning task. To classify a testing instance $\mathbf{x}^* \in \mathbf{X}^{te}$, we apply nearest neighbour matching of the projected testing instance $f(\mathbf{x}^*)$ against the vector representations of all the testing classes $g(y)$ (prototype):

$$\hat{y} = \arg\min_{y \in \mathbf{y}^{te}} \|f(\mathbf{x}^*) - g(y)\| \tag{5.13}$$

Distances in such embedding spaces have been shown to be best measured using the cosine metric [24, 139]. Thus we l_2 normalise each data point, making Euclidean distance effectively equivalent to cosine distance in this space.

5.2.1 Ameliorating Domain Shift by Post Processing

In the previous two sections we introduced two methods to improve the embedding f for ZSL. In this section we now discuss two post-processing strategies to further reduce the impact of domain shift.

Self-Training for Domain Adaptation

The domain shift induced by applying $f(\cdot)$ trained on $\mathbf{X}^{tr}$ to data of different statistics $\mathbf{X}^{te}$ means the projected data points $f(\mathbf{X}^{te})$ do not lie neatly around the corresponding class projections/prototypes $g(\mathbf{y}^{te})$ [22]. To ameliorate this domain shift, we explore transductive self-training to adjust unseen class prototypes to be more comparable to the projected data points. For each category prototype $g(y^*), y^* \in \mathbf{y}^{te}$ we search for the N_K^{st} nearest neighbours among the unlabelled testing instance projections, and re-define the adapted prototype $\tilde{g}(y^*)$ as the average of those N_K^{st} neighbours. Thus if $NN_K(g(y^*))$ denotes the set of K nearest neighbours of $g(y^*)$, we have:

$$\tilde{g}(y^*) := \frac{1}{N_K^{st}} \sum_{f(\mathbf{x}^*) \in NN_K(g(y^*))}^{N_K^{st}} f(\mathbf{x}^*) \tag{5.14}$$

The adapted prototypes $\tilde{g}(y^*)$ are now more directly comparable with the testing data for matching using Eq. (5.13).

An alternative explanation to self-training is all novel category prototypes are brought closer to the centre of all testing data. The parameter $\mathbf{K}$ determines how far the novel prototypes would travel. In an extreme case where $\mathbf{K} = n_u$ all prototypes would converge into a single point which is the centre of all testing data.

Hubness Correction

One practical effect of the ZSL domain shift was elucidated in [46], and denoted the 'Hubness' problem. Specifically, after the domain shift, there are a small set of 'hub' test-class prototypes that become nearest or K nearest neighbours to the majority of testing samples in the semantic space, while others are NNs of no testing instances. This results in poor accuracy and highly biased predictions with the majority of testing examples being assigned to a small minority of classes. We therefore explore the simple solutions proposed by [46] which takes into account the global distribution of zero-shot samples and prototypes. This method is transductive as with self-training and manifold-regression. Specifically, we considered two alternative approaches: *Normalised Nearest Neighbour* (NRM) and *Globally Corrected* (GC).

The NRM approach eliminates the bias towards hub prototypes by normalising the distance of each prototype to all testing samples prior to performing Nearest Neighbour classification as defined in Eq (5.13). More specifically, denote the distance between prototype y_j and testing sample $\{\mathbf{x}_i^*\}_{i=1\cdots n_u}$ as $d_{ij} = ||f(\mathbf{x}_i^*) - g(y_j)||$. We then l_2 normalise the distances between prototype y_j and all n_u testing samples in Eq (5.15). This normalised distance $\hat{d}_{ij}$ replaces the original distance d_{ij} for doing nearest neighbour matching in Eq. (5.13).

$$\tilde{d}_{ij} = d_{ij} / \sqrt{\sum_{i}^{n_u} d_{ij}^2} \tag{5.15}$$

The alternatively GC approach damps the effect of hub prototypes by using ranks rather than the original distance measures. We denote the function $Rank(y, \mathbf{x}_i^*)$ as the rank of testing sample $\mathbf{x}_i^*$ w.r.t the distance to y. Specifically, the rank function is defined as Eq (5.16) where $\mathbb{1}$ is the indicator function.

$$Rank(y, \mathbf{x}_i^*) = \sum_{\mathbf{x}_j^* \in \mathbf{X}^{te} \setminus \mathbf{x}_i^*} \mathbb{1}(||f(\mathbf{x}_j^*) - g(y)|| \leq ||f(\mathbf{x}_i^* - g(y))||) \tag{5.16}$$

The rank function always return an integer value between 0 and $|\mathbf{X}^{te}| - 1$. Thus the label of testing sample x_i^* can be predicted by Eq (5.17) in contrast to simple nearest by neighbour Eq (5.13).

$$\hat{y} = \arg\min_{y \in \mathbf{y}^{te}} Rank(y, \mathbf{x}_i^*) \tag{5.17}$$

Note, both strategies do not alter the ranking of testing samples w.r.t. each prototype. However, the ranking of prototypes w.r.t. each testing sample is altered thus potentially improves the quality of NN matching. Overall, due to the nature of a retrieval task which depends on the ranking of testing samples w.r.t. prototypes, the performance of retrieval task is not affected by the two hubness correction methods.

5.2.2 Transductive Setting

Of the four strategies introduced before, manifold regularization, self-training, and hubness correction assume access to the full set of unlabelled testing data, which is called the transductive setting [57, 22]. This assumption would be true in many real-world problems. Video repositories, e.g. YouTube, can process large batches of unlabelled videos uploaded by users. Transductive zero-shot methods can be used to tag batches automatically without manual annotation, or add a new tag to the ontology of an existing annotated set.

5.2.3 Multi-Shot Learning

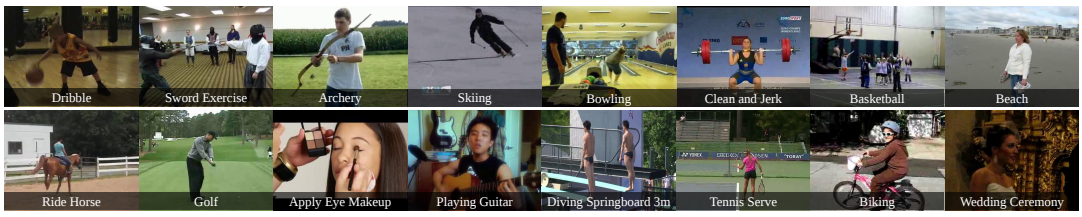
Although our focus is zero-shot learning, we also note that the semantic embedding space provides an alternative representation for conventional supervised learning. For multi-shot learning, we map all data instances $\mathbf{X}$ into the semantic space using projection $\mathbf{Z} = f(\mathbf{X})$, and then simply train SVM classifiers with linear kernel using the l_2 normalised projections $f(\mathbf{X})$ as data. In the testing phase, testing samples are projected into embedding space via the mapping $f(\mathbf{X})$ and categorised using the SVM classifiers.

5.3 Experiments

5.3.1 Datasets and Settings

Datasets

Experiments are performed on 5 popular contemporary action recognition and event detection datasets including A Large Human Motion Database (HMDB51) [4], UCF101 [5], Olympic Sports [7] and Columbia Consumer Video (CCV) [6]. HMDB51 is specifically created for human action recognition. It has 6766 videos from various sources with 51 categories of actions. UCF101 is an action recognition dataset of 13320 realistic action videos, collected from YouTube, with 101 action categories. Olympic Sports is collected from YouTube, and is mainly focused on sports events. It has 783 videos with 16 categories of events. CCV contains 9682 YouTube videos over 20 semantic categories. We illustrate some example frames in Figure 5.2. The action/event category names are presented in Table 5.2. We also evaluate USAA [139] – a subset of CCV specifically annotated with attributes – in order to facilitate comparison against attribute centric ZSL approaches. In addition to above action/event datasets, we also studied a large complex event dataset - TRECVID MED 2013. There are five components to the dataset including Event Kit training, Background training, test set MED, test set Kindred and Research Set. We use standard test set MED for zero-shot testing data and Event Kit as training data.



(a) HMDB51 (b) UCF101 (c) OlympicSports (d) *CCV*

Figure 5.2: Example frames for different action datasets.

Visual Feature Encoding

For each video we extract improved trajectory feature (ITF) descriptors [87] and encode them with Fisher Vectors (FV). We first compute ITF with 3 descriptors (HOG, HOF and MBH). We apply PCA to reduce the dimension of descriptors by half which results in descriptors with 198 dimensions in total. Then we randomly sample 256,000 descriptors from each of the 5 action/event datasets and learn a Gaussian Mixture Model with 128 components from the combined training

Table 5.2: Category names of each human action dataset.

Dataset	Category Names
HMDB51	brush_hair, cartwheel, catch, chew, clap, climb, climb_stairs, dive, draw_sword, dribble, drink, eat, fall_floor, fencing, flic_flac, golf, handstand, hit, hug, jump, kick, kick_ball, kiss, laugh, pick, pour, pullup, punch, push, pushup, ride_bike, ride_horse, run, shake_hands, shoot_ball, shoot_bow, shoot_gun, sit, situp, smile, smoke, somersault, stand, swing_baseball, sword, sword_exercise, talk, throw, turn, walk, wave
UCF101	Apply Eye Makeup, Apply Lipstick, Archery, Baby Crawling, Balance Beam, Band Marching, Baseball Pitch, Basketball Shooting, Basketball Dunk, Bench Press, Biking, Billiards Shot, Blow Dry Hair, Blowing Candles, Body Weight Squats, Bowling, Boxing Punching Bag, Boxing Speed Bag, Breaststroke, Brushing Teeth, Clean and Jerk, Cliff Diving, Cricket Bowling, Cricket Shot, Cutting In Kitchen, Diving, Drumming, Fencing, Field Hockey Penalty, Floor Gymnastics, Frisbee Catch, Front Crawl, Golf Swing, Haircut, Hammer Throw, Hammering, Handstand Pushups, Handstand Walking, Head Massage, High Jump, Horse Race, Horse Riding, Hula Hoop, Ice Dancing, Javelin Throw, Juggling Balls, Jump Rope, Jumping Jack, Kayaking, Knitting, Long Jump, Lunges, Military Parade, Mixing Batter, Mopping Floor, Nun chucks, Parallel Bars, Pizza Tossing, Playing Guitar, Playing Piano, Playing Tabla, Playing Violin, Playing Cello, Playing Daf, Playing Dhol, Playing Flute, Playing Sitar, Pole Vault, Pommel Horse, Pull Ups, Punch, Push Ups, Rafting, Rock Climbing Indoor, Rope Climbing, Rowing, Salsa Spins, Shaving Beard, Shotput, Skate Boarding, Skiing, Skijet, Sky Diving, Soccer Juggling, Soccer Penalty, Still Rings, Sumo Wrestling, Surfing, Swing, Table Tennis Shot, Tai Chi, Tennis Swing, Throw Discus, Trampoline Jumping, Typing, Uneven Bars, Volleyball Spiking, Walking with a dog, Wall Pushups, Writing On Board, Yo Yo
Olympic Sports	basketball layup, bowling, clean and jerk, discus throw, hammer throw, high jump, javelin throw, long jump, diving platform 10m, pole vault, shot put, snatch, diving springboard 3m, tennis serve, triple jump, vault
CCV	Basketball, Baseball, Soccer, IceSkating, Skiing, Swimming, Biking, Cat, Dog, Bird, Graduation, Birthday, WeddingReception, WeddingCeremony, WeddingDance, MusicPerformance, NonmusicPerformance, Parade, Beach, Playground

descriptors. Finally the dimension of FV encoded feature is equal to $d_x = 2 \times 128 \times 198 = 50688$. The visual feature for TRECVID MED 2013 dataset was extracted using ITF with HOG and MBH descriptors encoded with Fisher Vectors. We use the FV encoded feature provided by [23].

Semantic Embedding Space

We adopted the skip-gram neural network model [24] trained on the Google News dataset (about 100 billion words). This neural network can then encode any of approximately 3 million unique worlds as a $d_z = 300$ dimension vector.

5.3.2 Zero-Shot Learning on Actions and Events

Data Split

Because there is no existing zero-shot learning evaluation protocol for most existing action and event datasets we propose our own splits. We first propose a 50/50 category split for all datasets. Visual-to-semantic mappings are trained on the 50% training categories, and the other 50% are held out unseen for testing time. We randomly generate 50 independent splits and take the mean accuracy and standard deviation for evaluation. Among the 50 splits, all categories are evaluated

as testing classes, and the frequency is evenly distributed.

Evaluation of Components

To evaluate the efficacy of each component we considered an extensive combination of blocks including manifold regularizer, self-training, hubness correction and data augmentation. Specifically we evaluated the following options for each component.

- **Data Augmentation:** Using only within target dataset training data (✗) to learn the embedding $f(\mathbf{x})$, or also borrowing data from the auxiliary datasets (✓). (Section 5.1.2). For each of the four datasets HMDB51, UCF101, Olympic Sports and CCV, the other three datasets are treated as the auxiliary sets. Note, there are overlapping categories between the auxiliary and target sets in the sense of exact name match. For instance, the action class *Biking* exists in both UCF101 and CCV. To avoid violating the zero-shot assumption we exclude these exact matching classes in the auxiliary set. However, we consider that semantic overlaps, e.g. *Biking* in UCF101 and *Ride Bike* in HMDB51, should not be excluded because recognising such paraphrase of action category is the problem to be solved by zero-shot learning and exploiting such semantic relatedness is unique to word-vector embedding approach.

- **Embedding:** We compare ridge regression (RR) with manifold regularized ridge regression (MR) (Section 5.1.2).

- **Self Training:** With (✓) or without (✗) self-training before matching (Section 5.2.1).

- **Matching Strategy:** We compare conventional NN matching (NN) Eq. (5.13) versus Normalised Nearest Neighbour (NRM) Eq. (5.15) and Globally Corrected (GC) matching Eq. (5.17) (Section 5.2.1). Note that the hubness correction methods (NRM and GC) do not change retrieval performance. Therefore, NN/ NRM/ GC do not perform differently on OlympicSports and CCV.

- **Transductive:** (Trans) Indicating whether the combination of components is transductive (✓) or not (✗). The former requires the access to unlabelled testing data.

Based on this breakdown of components, we note that the condition (✗-RR-✗-NN-✗) is roughly equivalent to the methods in [26] and [141], and the conditions (✗-RR-✗-GC-✓, ✗-RR-✗-NRM-✓) are roughly equivalent to [46]. As we note Self-Training brings the biggest

contribution to performance boost we single out the evaluation of ST in Table 5.4 and present the results for rest components in Table 5.3.

Metrics

HMDB, UCF and USAA are classification benchmarks, so we report average accuracy metric. Olympic Sports and CCV are detection benchmarks, so we report mean average precision (mAP) metrics. We note that because distance normalisation (NRM) does not change the relative rank of testing instances w.r.t. testing class, there is no difference between NRM and NN for mAP. Therefore, we insert a '$-$' for Match-NRM on Olympic Sports and CCV. The performance for these '$-$' is the same as their NN counterparts.

Table 5.3: Evaluation of the contribution of individual component (average % accuracy $\pm$ standard deviation for HMDB51, UCF101 and USAA and mean average precision $\pm$ standard deviation for Olympic Sports and CCV). All '$-$' indicate no difference in performance between NN and NRM.

Model	Match	Data Aug	Trans	HMDB51	UCF101	Olympic Sports	CCV	USAA
RR	NN	X	X	14.5±2.7	11.7±1.7	35.7±8.8	20.7±3.0	29.5±5.5
MR	NN	X	✓	15.9±3.1	12.9±2.2	37.7±9.5	21.4±3.0	29.8±4.0
RR	GC	X	✓	15.3±2.7	13.5±1.8	35.7±8.8	20.7±3.0	26.1±6.7
RR	NRM	X	✓	16.1±2.7	13.9±1.5	-	-	28.6±7.2
MR	NRM	X	✓	18.0±3.2	15.6±2.0	-	-	28.2±5.4
RR	NN	✓	X	20.4±2.9	15.7±1.6	38.6±7.5	30.3±3.9	28.2±4.6
RR	NRM	✓	✓	21.0±2.7	18.5±1.7	-	-	35.6±2.6
MR	NN	✓	✓	20.6±2.9	17.2±1.6	41.1±7.7	30.4±3.9	30.3±4.9

Experimental Results

We make the following observations from the results in Table 5.3: (i) The simplest approach of directly mapping features to the embedding space (X-RR-X-NN-X [26, 141]) works reasonably well suggesting that semantic space is effective as a representation and supports ZSL. (ii) Manifold regularization reliably improves performance compared to conventional ridge regression by reducing the domain shift through considering the unlabelled testing data (transductive learning). (iii) Data augmentation also significantly improves the results by providing a more representative sample of training data for learning the embedding. (iv) According to Table 5.4, self-training [22] post-processing improves results at testing time, and this is complementary with our proposed manifold regularization and data augmentation.

Table 5.4: Evaluation of the contribution of Self-Training component.

Model	Match	ST	Data Aug	HMDB51	UCF101	Olympic Sports	CCV	USAA
RR	NN	X	X	14.5±2.7	11.7±1.7	35.7±8.8	20.7±3.0	29.5±5.5
RR	NN	✓	X	17.0±3.1	15.9±2.3	37.3±9.1	21.7±3.2	30.2±5.2
MR	NRM	X	X	18.0±3.2	15.6±2.0	-	-	28.2±5.4
MR	NRM	✓	X	19.1±3.8	18.0±2.7	-	-	31.6±3.2
RR	NN	X	✓	20.4±2.9	15.7±1.6	38.6±7.5	30.3±3.9	28.2±4.6
RR	NN	✓	✓	23.6±3.7	21.2±2.4	42.0±8.2	**33.8±4.7**	42.8±8.7
MR	NN	X	✓	20.6±2.9	17.2±1.6	41.1±7.7	30.4±3.9	30.3±4.9
MR	NN	✓	✓	23.5±3.9	20.6±2.4	**43.2±8.3**	33.0±4.8	41.2±9.7
MR	NRM	✓	✓	**24.1±3.8**	**22.1**±2.5	-	-	**43.3±10.9**

Comparison With State-of-the-Art

In addition to the above variants of our framework, we also evaluate the following state-of-the-art approaches to ZSL on action recognition tasks. As both word-vector embedding and manually labelled attributes are widely studied in the literature of zero-shot learning, we compare our approach using both word-vector and attribute semantic embedding with the state-of-the-art models. Attribute embedding is only evaluated on UCF, Olympic Sports and USAA where attributes are available.

Word-Vector Embedding: For word-vector embedding, we evaluate three alternative models:

1. **Structured Joint Embedding (SJE)** We use the code of [29] with FV encoded visual feature to evaluate the performance on all 5 datasets. The SJE model employs bilinear ranking to ensure relevant labels (word-vectors) are ranked higher than irrelevant labels.

2. **Convex Combination of Semantic Embeddings (ConSE)** We implement the ConSE model [43] with the same FV encoded feature and evaluate on all 5 datasets. The ConSE model firstly trains classifiers for each known category $p(y_j|\mathbf{x})$. Given testing visual data $\mathbf{x}$, the semantic embedding of visual data is synthesised by a linear combination of known category embeddings as $f(\mathbf{x}) = \sum_{j=1}^{T} p(y_j|\mathbf{x})\mathbf{z}_j$ where T is the top T known classes.

3. **Support Vector Embedding (SVE)** This model [190] learns the visual-to-semantic mapping via support vector regression. Performance is reported on HMDB51 and UCF101 datasets.

Attribute Embedding: In addition to word-vector embedding based methods, we also compare against existing state-of-the-art models using attribute embeddings. To enable direct comparisons with the same embedding, we carry out experiments for our approach with attribute embedding as well (although in this setting our data augmentation cannot be applied). Specifically, we compare the following methods:

1. **Direct Attribute Prediction (DAP)** We implement the method of [39], but using the same FV encoded visual features and linear kernel SVM attribute classifiers $p(\mathbf{a}|\mathbf{x})$. Recognition is then performed based on attribute posteriors and manually specified attribute descriptor $p(\mathbf{a}|y)$.

2. **Indirect Attribute Prediction (IAP)** [39]. This differs from DAP by learning a per-category classifier $p(y|\mathbf{x})$ from training data first and then use the training category attribute-prototype dependency $p(\mathbf{a}|y)$ to obtain attribute estimator $p(\mathbf{a}|\mathbf{x})$.

3. **Human Actions by Attributes (HAA)** [20]. We reproduce a simplified version of this model which exploits the manually labelled attributes $\{a_m\}$ for zero-shot learning. Similar to DAP, a binary SVM classifier is trained per attribute. In the testing phase, each testing sample is projected into attribute space and then assigned to the closest testing/unknown class based on cosine distance to the class prototype (NN).

4. **Propagation Semantic Transfer (PST)** [191] and [192]. Label propagation is adopted in this approach to adjust the initial predictions of DAP. Specifically, a KNN graph is constructed in the attribute embedding space and a smoothed solution is obtained transductively by semi-supervised label propagation [193].

5. **Multi-Modal Latent Attribute Topic Model (M2LATM)** [139]. It exploits both user-defined and discovered latent attributes to facilitate zero-shot learning. This model fuses multiple features – static (SIFT), motion (STIP) and audio (MFCC), and thus has an advantage compared to other methods evaluated that use vision alone. We report the results on USAA from [139].

6. **Transductive Multi-View Bayesian Label Propagation (TMV-BLP) [27]**. This model builds a common space for multiple embeddings. It combines attribute and word-vectors, and applies bayesian label propagation to infer the category of testing instances. It evaluated on USAA dataset with SIFT, STIP and MFCC features.

7. **Transductive Multi-View Hypergraph Label Propagation (TMV-HLP) [22]**. An improved version of TMV-BLP. A distributed hypergraph was adopted to replace the local neighbourhood graph in [27].

8. **Unsupervised Domain Adaptation (UDA)**. The UDA model [145] learns dictionary on auxiliary data and adapts it to the target data as a constraint on the target dictionary rather than blindly using the same dictionary.

Mixed Embedding: We refer to exploiting attribute and word-vector embeddings jointly as studied by [22] and [29]. Although multi-view embedding is not the focus of this work, we evaluate our model with a simple concatenation of attribute and word-vector embeddings. Three alternatives are compared including TMV-BLP [27], UDA [145] and TMV-HLP [22].

Method Properties: We indicate the nature of each approach with four parameters. **DA** - if data augmentation is applied. **Trans** - whether the approach requires transductive access to testing data. **Embed** - what semantic embedding is used. Embed-A, Embed-W and Embed A+W indicate attribute, word vector, and both attribute+word vector embeddings respectively. **Feat** - What visual feature is used. FV indicates Fisher vector encoded dense trajectory feature; BoW indicates bag of words encoded dense trajectory feature; and SMS indicates joint SIFT, MFCC and STIP feature.

Experimental Results

The full results are presented in Table 5.5, from which we draw the following conclusions: (i) Our non-transductive model (RR) is strong compared with alternative models with either word-vector embedding or attribute embedding. For example, our RR model is able to beat SJE and ConSE in UCF101, CCV and USAA with word-vector embedding and beat DAP, IAP and HAA in Olympic Sports and USAA. (ii) With transductive access to testing data, our model MR-X-✓-W is better than most alternative models with word-vector and competitive against models with attribute embedding. (iii) The overall combination of all components, manifold regularized regression (MR), Data Augmentation (DA) and Self-training and hubness (Trans), with word-vector embedding (MR-✓-✓-W) can yield very competitive performance. Depending on the dataset, our overall model is comparable or significantly better than the attribute-centric methods, e.g. UCF101. (iv) With mix-embedding (A+W) our model is still very competitive against existing ZSL approaches and outperform TMV-BLP, UDA and TMV-HLP. Apart from the above observations

Table 5.5: Comparison with state-of-the-art approaches to ZSL. Both attribute and word-vector embeddings are studied for fair comparison. * performances are estimated from Fig. 2 (a) $\Gamma(X+V)$ and $\Gamma(X+A)$ respectively in [27]. ** performances are estimated from Fig. 5 (c) $\Gamma(X+V)$ and $\Gamma(X+A)$ respectively in [22]. N/A indicates not available due to the absence of attribute annotation or not reported by the original work.

Model	DA	Trans	Embed	Feat	HMDB51	UCF101	Olympic Sports	CCV	USAA
Random Guess	X	X	X	X	4.0	2.0	12.5	10.0	25.0
RR (Ours)	X	X	W	FV	14.5±2.7	11.7±1.7	35.7±8.8	20.7±3.0	29.5±5.5
MR (Ours)	X	✓	W	FV	19.1±3.8	18.0±2.7	38.6±10.6	22.5±3.4	31.6±3.2
MR (Ours)	✓	✓	W	FV	**24.1±3.8**	**22.1±2.5**	**43.2±8.3**	**33.0±4.8**	**43.3±10.9**
SJE [29]	X	X	W	FV	12.0±2.6	9.3±1.7	34.6±7.6	16.3±3.1	21.3±0.6
ConSe [43]	X	X	W	FV	15.0±2.7	11.6±2.1	36.6±9.0	20.7±3.1	28.2±4.8
TMV-BLP [27]*	X	✓	W	SMS	N/A	N/A	N/A	N/A	41.0
TMV-HLP [22]**	X	✓	W	SMS	N/A	N/A	N/A	N/A	43.0
SVE [190]	X	X	W	BoW	12.9±2.3	11.0±1.8	N/A	N/A	N/A
RR (Ours)	X	X	A	FV	N/A	12.6±1.8	51.7±11.3	N/A	44.2±13.9
MR (Ours)	X	✓	A	FV	N/A	**20.2±2.2**	**53.5±11.9**	N/A	**51.6±10.0**
DAP [39]	X	X	A	FV	N/A	15.2±1.9	44.4±9.9	N/A	37.9±5.9
IAP [39]	X	X	A	FV	N/A	15.6±2.2	44.0±10.7	N/A	31.7±1.6
HAA [20]	X	X	A	FV	N/A	14.3±2.0	48.3±10.2	N/A	41.2±9.8
PST [191]	X	✓	A	FV	N/A	15.3±2.2	48.6±11.0	N/A	47.9±10.6
M2LATM [139]	X	✓	A	SMS	N/A	N/A	N/A	N/A	41.9
TMV-BLP [27]*	X	✓	A	SMS	N/A	N/A	N/A	N/A	40.0
TMV-HLP [22]**	X	✓	A	SMS	N/A	N/A	N/A	N/A	42.0
UDA [145]	X	✓	A	FV	N/A	13.2±1.9	N/A	N/A	N/A
MR (Ours)	X	✓	A+W	FV	N/A	**20.8±2.3**	**53.2±11.6**	N/A	**51.9±10.1**
TMV-BLP [27]	X	✓	A+W	SMS	N/A	N/A	N/A	N/A	47.8
UDA [145]	X	✓	A+W	FV	N/A	14.0±1.8	N/A	N/A	N/A
TMV-HLP [22]	X	✓	A+W	SMS	N/A	N/A	N/A	N/A	50.4

we note that the ZSL performance variance is relatively high, particularly in Olympic Sports and USAA datasets. This is because specific choice of train/test classes in ZSL matters more than specific choice of train/test instances in conventional supervised learning, e.g. in olympic sports there are highly related categories 'high jump' - 'long jump' and 'diving platform 10m' - 'diving springboard 3m'. Recognition performance is higher when these pairs are separated in training and testing, and lower if they are both in testing. This issue is explored further in section 5.3.5.

5.3.3 Zero-Shot Learning of Complex Events

In this section, we experiment on the more challenging complex event dataset - TRECVID MED 2013.

Data Split

We study the 30 classes of the MED test set, holding out the 20 events specified by the 2013 evaluation scheme for zero-shot recognition, and training on the other 10. We train on the total 1611 videos in Event Kit Train (160 per event in average) and test on the 27K examples in MED test, of which only about 1448 videos are the 20 events to be detected. This is different to the standard TRECVID MED 2013 0EK in which concept detectors are trained on the Research Set [23, 146, 101]. This experimental design is chosen because we want to exploit *only* per-category annotation (event name) as semantic supervision, rather than requiring the per-video sentence annotation used in the Research Set which is very expensive to collect. This work is the first attempt to address TRECVID MED 2013 with only event name annotation. However, it means we use fewer training videos (1611) compared to the 10K video Research Set. Thus our results are not comparable to existing TRECVID MED 2013 0EK benchmark results, because we use vastly less training data.

Table 5.6: Events for training visual-to-semantic regression.

ID	Event Name	ID	Event Name
E001	Attempting a board trick	E002	Feeding an animal
E003	Landing a fish	E004	Wedding ceremony
E005	Working on a woodworking project	E016	Doing homework or studying
E017	Hide and seek	E018	Hiking
E019	Installing flooring	E020	Writing

Baselines

We compare 5 alternative baselines for TRECVID MED zero-shot event detection.

1. **Random Guess** - Randomly rank the candidates.
2. **NN** (X-RR-X-NN-X). Rank videos with l_2 distance in the semantic space.
3. **NN + ST** (X-RR-✓-NN-✓). Adjust prototypes with self-training.
4. **Manifold** (X-MR-X-NN-✓). Add manifold regularization term in the visual-to-semantic regression model.

Table 5.7: Events for testing zero-shot event detection.

ID	Event Name	ID	Event Name
E006	Birthday party	E007	Changing a vehicle tire
E008	Flash mob gathering	E009	Getting a vehicle unstuck
E010	Grooming an animal	E011	Making a sandwich
E012	Parade	E013	Parkour
E014	Repairing an appliance	E015	Working on a sewing project
E021	Attempting a bike trick	E022	Cleaning an appliance
E023	Dog show	E024	Giving directions to a location
E025	Marriage proposal	E026	Renovating a home
E027	Rock climbing	E028	Town hall meeting
E029	Winning a race without a vehicle	E030	Working on a metal crafts project

5. **Manifold + ST** (✗-MR-✓-NN-✓) - manifold regularization regression with self-training.

We were not able to investigate data augmentation for TRECVID due to the different feature encoding from the other action datasets.

We present the performance of zero-shot learning on TRECVID MED 2013 in Figure 5.3 and Table 5.8. Figure 5.3 reports the performance of 4 alternative models and random guess baseline in detecting 20 events in mean average precision (mAP) and the average over all events (Average). Compared to Random Guess (0.28%), our direct embedding approach (NN) is effective at zero-shot video detection. Self-Training and Manifold Regularization further improve the performance. Table 5.8 puts the results in broader context by summarising them in terms of absolute performance.

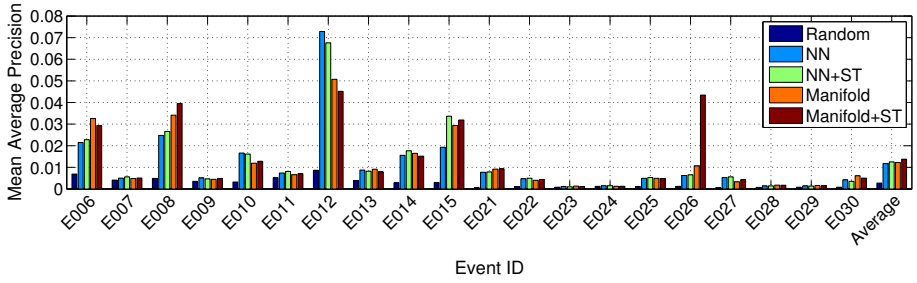


Figure 5.3: zero-shot performance on TRECVID MED 2013 measured in mean average precision (mAP).

5.3.4 Zero-Shot Qualitative Visualisation

In this section we illustrate qualitatively the effect of our contributions on the resulting embedding space matching problem. For visualisation, we randomly sample 5 testing classes from

Table 5.8: Event detection performance on TRECVID MED 2013. mAP across 20 events to be detected.

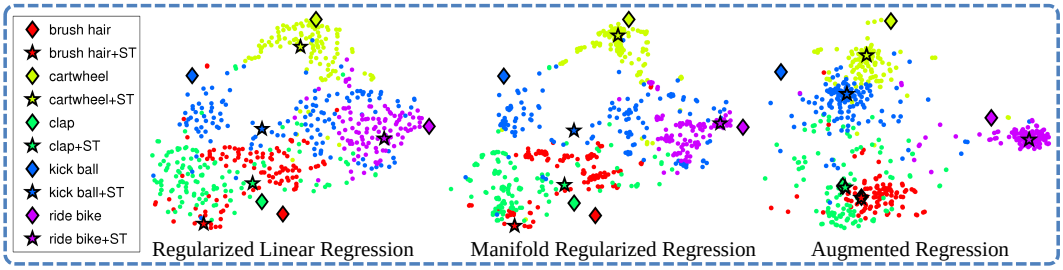
Embed	ST	Match	Average mAP
RR	X	NN	1.18%
RR	✓	NN	1.25%
MR	X	NN	1.22%
MR	✓	NN	1.38%
Random Guess			0.28%

HMDB51 and project all samples from these classes into the semantic space by (1) conventional ridge regression; (2) manifold regularized regression and (3) manifold regularized ridge regression with data augmentation. The results are visualised in 2D in Figure 5.4 with t-SNE [194]. Three sets of testing classes are presented for diversity. Data instances are shown as dots, prototypes (class name projections) as diamonds, and self-training adapted prototypes as stars. Colours indicate category.

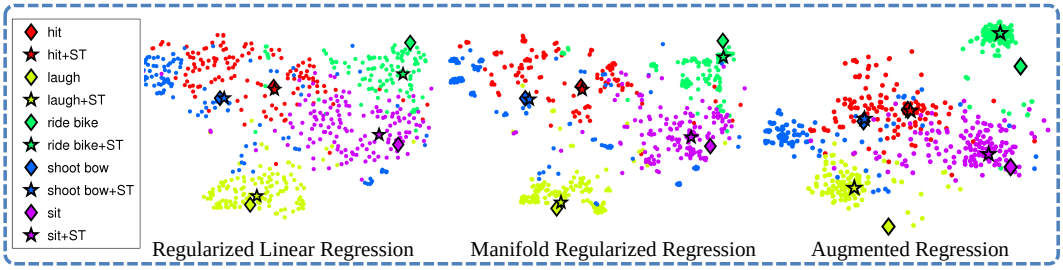
There are three main observations from Figure 5.4: (1) manifold regularized regression yields better visual-to-semantic projections as instances of the same class tend to form tighter clusters. This is due to the constraint of preserving the manifold structure from the visual feature space; (2) data augmentation yields an even more accurate projection of unseen data, as instances are projected closer to the prototypes and classes are more separable; and (3) self-training is effective as the adapted prototypes (stars) are closer to the centre of the corresponding samples (dots) than the original prototypes (diamonds). These observations illustrate the mechanism of our ZSL accuracy improvement on conventional approaches.

5.3.5 Understanding ZSL and Predicting Transferrability

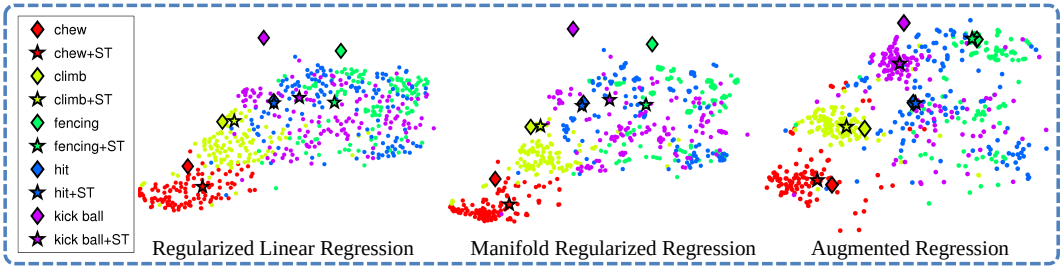
In this section we present further insight into considerations on what factors will affect the efficacy of ZSL, through a category-level analysis. The basic assumption of ZSL is that the embedding $f(\mathbf{x})$ trained on known class data, will also apply to testing classes. As we have discussed throughout this study, this assumption is stretched to some extent due to the disjoint training and testing category sets. This leads us to investigate how zero-shot performance depends on the specific choice of training classes and their relation to the held out testing classes.



(a) Category set 1



(b) Category set 2



(c) Category set 3

Figure 5.4: A qualitative t-SNE illustration of ZSL with semantic space representation for random testing class subsets (a), (b) and (c). Variants: ridge regression, manifold regression and data augmented manifold regression. Dots indicate instances, colour categories, and star/diamond show category prototypes with/without self-training.

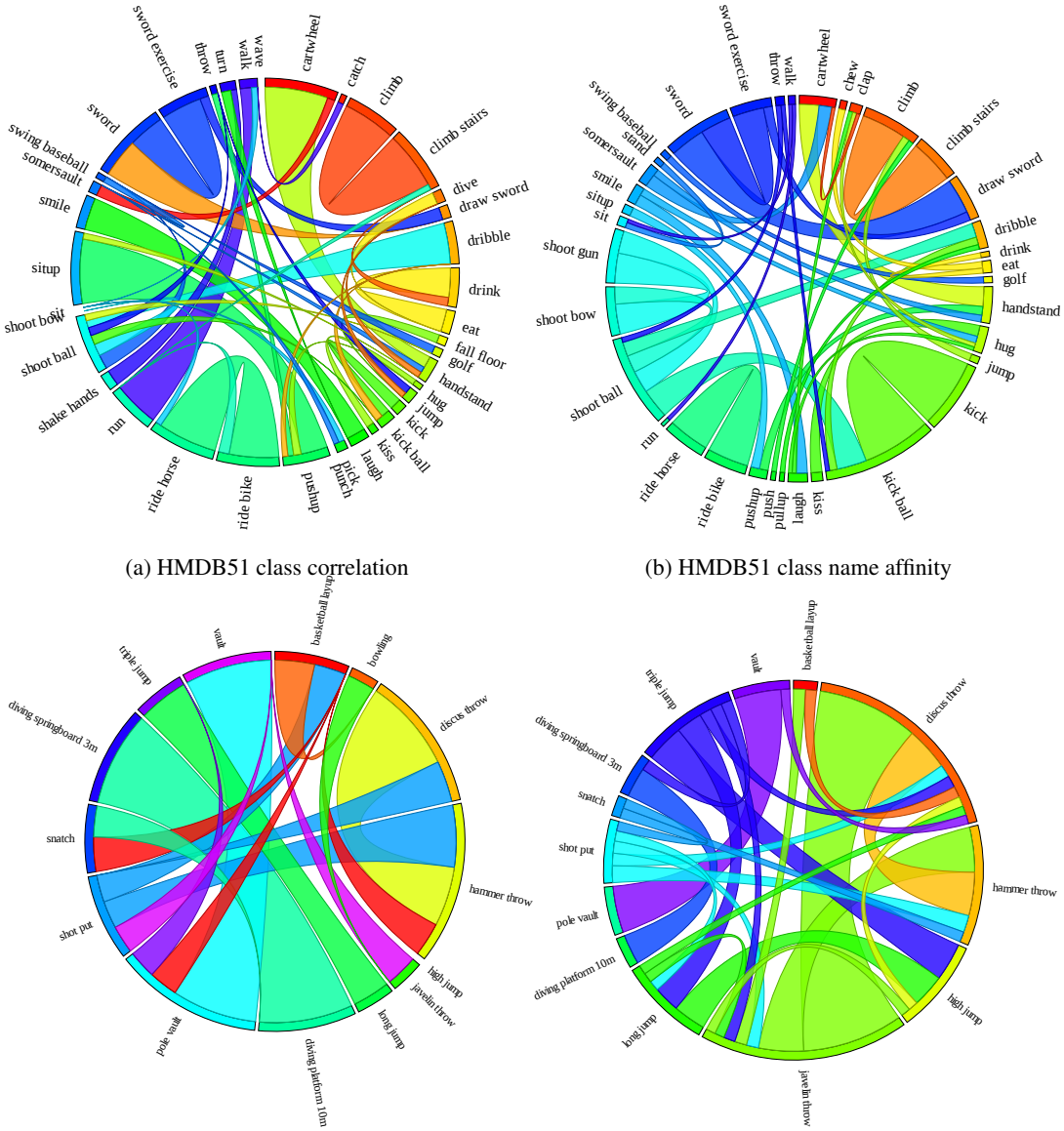
Impact of Training Class Choice on Testing Performance

We first investigate whether there are specific classes which, if included as training data, significantly impact testing class performance. To study this, we compute the correlation between training class inclusion and testing performance. Specifically, we consider a pair of random variables $\{b_i^{tr}, e_j^{te}\}$ where b_i^{tr} is a binary scalar indicating if the ith class is in the training set and e_j is the recognition accuracy of the jth testing class. We compute the correlation $corr(i,j)$ between every pair of variables over the 50 random splits:

$$corr(i,j) = \frac{\mathbb{E}[(b_i^{tr} - \overline{b_i^{tr}})(e_j^{te} - \overline{e_j^{te}})]}{var(b_i^{tr})var(e_j^{te})}. \tag{5.18}$$

We use chord diagrams to visualise the relation between categories in Fig 5.5(a). The strength of positive cross-category correlation is indicated by the width of the bands connecting the cate-

gories on the circle, i.e., a wide band indicates inclusion of one category as training data facilitates the zero-shot recognition of the. Due to the large number of categories we apply two preprocessing steps before plotting: (1) convert all correlation coefficients to positive value by exponentially power scaling the correlation coefficient; and (2) remove highly negative correlated pairs to avoid clutter.



(a) HMDB51 class correlation

(b) HMDB51 class name affinity

(c) Olympic Sports class correlation

(d) Olympic Sports class name affinity

Figure 5.5: Chord Diagram to illustrate the category correlation discovered from zero-shot recognition experiments. (a) and (c) illustrate the correlation discovered from 50 random split zero-shot experiments; (b) and (d) illustrate the class name affinity in word-vector embedding space measured as cosine similarity.

We can make several qualitative observations from the chord diagrams. The class correlation captures the dependence of category B's recognition rate on category A's presence in the training

set. So for instance for A=*ride horse* and B=*ride bike*, Figure 5.5(a) shows that we would expect high recognition accuracy of *ride horse* if *ride bike* is present in training set and vice versa. However while *cartwheel* supports the recognition of *handstand*, the reverse is not true.

Cross-Class Transferability Correlates with Word-Vector Similarity

We next investigate the affinity between class names' vector representations, and cross-class transferability. Class name affinities are shown in Fig 5.5(b) as chord diagrams. Visually there is some similarity to the cross-class transferability presented in Fig 5.5(a). To quantify this connection between transfer efficacy and classname relatedness, we vectorise the correlation (Fig 5.5(a)) and class name affinity (Fig 5.5(b)) matrices (51×51) into 2601 dim vectors and then compute the correlation coefficients between the two vectors. The correlation is 0.548, suggesting that class name relatedness and efficacy for ZSL are indeed connected. This is to say, if class A is present in training set and class B in testing set, and A has high affinity with B in word-vector distance measure, we could expect high performance in recognising class B.

To qualitatively illustrate this connection, we list the top 10 positively correlated category pairs in Table 5.9. Here the correlation of action 1 being in training and action 2 in testing is given as *Fwd Corr*, with *Back Corr* being the opposite. The affinity between category names are given as *WV Aff* which is defined as percentile rank of word-vector distance (closer to 1 means more similar). Clearly highly correlated categories have higher word-vector similarity.

Table 5.9: Top 10 positive correlated class pairs

Action 1	Action 2	Fwd Corr	Back Corr	WV Aff
climb stairs	climb	0.94	0.92	0.98
ride horse	ride bike	0.95	0.91	0.98
situp	pushup	0.96	0.79	0.91
sword exercise	sword	0.87	0.85	0.98
handstand	cartwheel	0.62	0.96	0.97
eat	drink	0.75	0.81	0.96
smile	laugh	0.82	0.72	0.97
walk	run	0.61	0.90	0.96
shoot ball	dribble	0.52	0.87	0.97
sword	draw sword	0.86	0.45	0.98

Although zero-shot transfer overall is effective, there are also some individual negative cor-

relations. We illustrate the distribution of positive and negative transfer outcomes in Figure 5.6. Here we sort all the class pairings into ten bins by their name affinity and plot the resulting histogram (blue bars). Clearly the majority of pairs have low classname affinity. For each bin of class-pairs, we also compute their average correlation defined in Eq 5.18 (Figure 5.6, red line). There are a few observations to be made: (1) class name affinity is clearly related to positive correlation: the correlation (red line) goes up significantly for high-affinity class pairs; (2) there are a relatively small number of category pairs that account for the high positive correlation outcomes (low blue bars to the right). This suggests that overall ZSL efficacy is strongly impacted by the presence of key supporting classes in the training set; and (3) there are a larger number of category pairs which exhibit negative transferability (red correlation is negative around affinity of 0.2). However negative transfer effects are quantitatively weak compared to positive transfer (red correlation line gets only weakly negative but strongly positive).

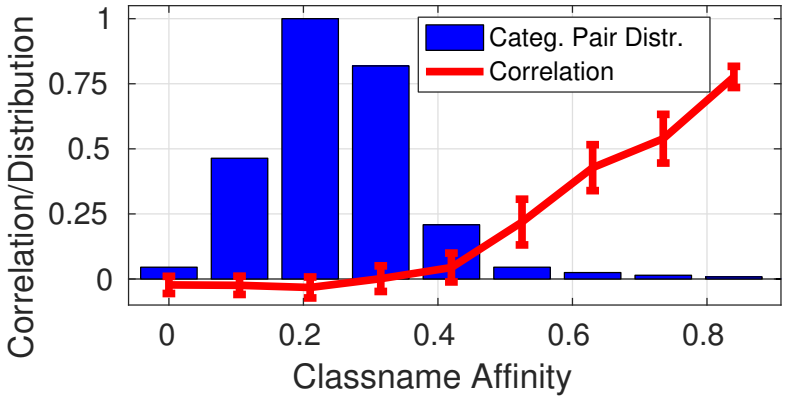


Figure 5.6: The connection between transfer efficacy and classname affinity: Illustrated by class correlation v.s. class name affinity.

Predicting Transferability

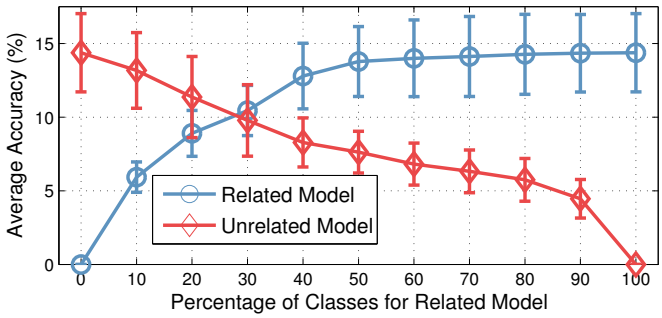
Based on the previous observations we hypothesize that class name affinity is predictive of ZSL performance, and may provide a guide to selecting a good set of training classes to maximise ZSL efficacy. We formally define the problem as given fixed testing categories $\{y_j|y_j \in \mathbf{y}^{te}\}$, we find the $S\%$ subset of training categories $\{y_i|y_i \in \mathbf{y}^{tr}\}$ which maximises the performance of recognising testing classes based on their affinity to the testing classes. We first of all explore three alternative (point-to-set) distances to measure the affinity of each training class y_i to the set of testing classes $\{y_j|y_j \in \mathbf{y}^{te}\}$, specifically the maximal/mean/minimal class name affinity:

$$R(y_i, \mathbf{y}^{te}) = \max_{y_j \in \mathbf{y}^{te}} (1 - ||g(y_i) - g(y_j||))$$
$$R(y_i, \mathbf{y}^{te}) = \operatorname*{mean}_{y_j \in \mathbf{y}^{te}} (1 - ||g(y_i) - g(y_j||)) \tag{5.19}$$
$$R(y_i, \mathbf{y}^{te}) = \min_{y_j \in \mathbf{y}^{te}} (1 - ||g(y_i) - g(y_j||))$$

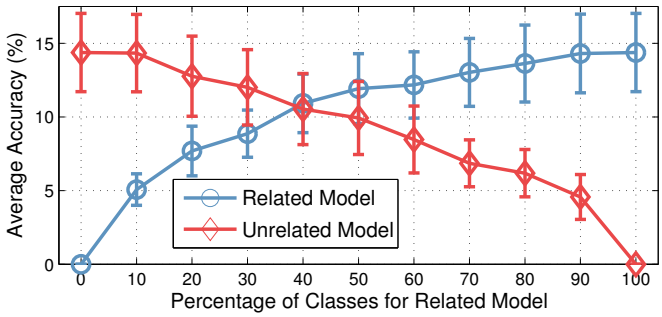
These metrics provide a means to quantify the relevance of any potential training class to the testing set. We explore their ability to predict transferability and hence construct a good training set for a particular set of testing classes.

For this experiment, we use HMDB51 with the same 50 random splits introduced in Section 5.3.2. Keeping the testing sets fixed, we train two alternative models based on different subsets of each training split. Specifically: (1) *Related Model* selects the top $S\%$ most related training classes (high affinity measure by $R(y_i, \mathbf{y}^{te})$) to the testing set defined by relatedness measure in Eq. (5.19) in order to learn the mapping; while (2) *Unrelated Model* selects the most $100 - S\%$ unrelated. Figure 5.7 shows the performance of both models as S varies between 0 and 100, where *Related* selects the top $S\%$ and *Unrelated* the bottom $100 - S\%$. Note that when $S = 0\%$ and $S = 100\%$ the *Unrelated* and *Related* models both select all training classes. Both are then equivalent to the standard ZSL model RR-NN-X-X introduced in Table 5.3. We illustrate the performance of both models and three alternative training-to-testing affinity measures in Figure (5.7).

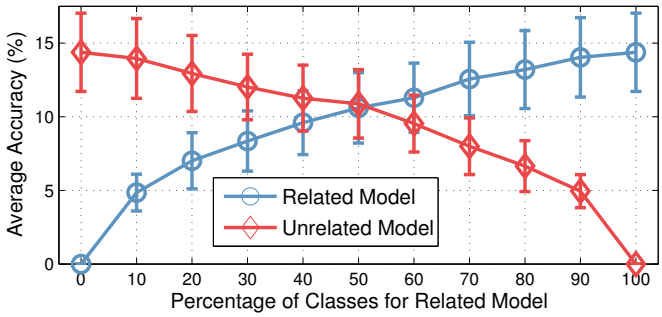
The main observations are as follows: (1) a crossover happens at 30% for maximal class name affinity, which means the model learned on the 30% subset of related training classes outperforms the model learned on the much larger 70% of unrelated classes; (2) the maximal class name affinity is most predicative on the efficacy of zero-shot learning as, firstly, the crossover point is the left most among all three alternative strategies, and, secondly, at the equal data point (50%) the related model most clearly outperforms the unrelated model; and (3) for maximal affinity, as more classes are included the related model increases in performance more rapidly than the unrelated one, and saturates after the top 50% are included. Both of these observations demonstrate that the related classes are more valuable than the unrelated classes (as the crossover is to the left of 50%), and that class name affinity of the training to testing set is predictive of the efficacy at testing time.



(a) Maximal class name affinity



(b) Mean class name affinity



(c) Minimal class name affinity

Figure 5.7: Testing the ability to predict ZSL class transferability by class name affinity: A comparison of models selecting related versus un-related classes as training data.

5.3.6 Unbalanced Test Set

Transductive strategies have been studied by many existing works [22, 46], however none of these works have ever studied the assumptions of test set for successful transductive ZSL. In particular, we note that, in zero-shot scenarios, testing categories could be highly imbalanced. How does the transductive strategies generalise to unbalanced test set remains an untouched problem. To verify this aspect, we carry out a particular experiment. Specifically, we experiment on the first split of HMDB51 and randomly subsample $P\%$ testing data from each of the first 12 testing categories for ZSL evaluation. We illustrate the distribution of testing videos per category for $P = 10, 50, 90$

in Figure 5.8.

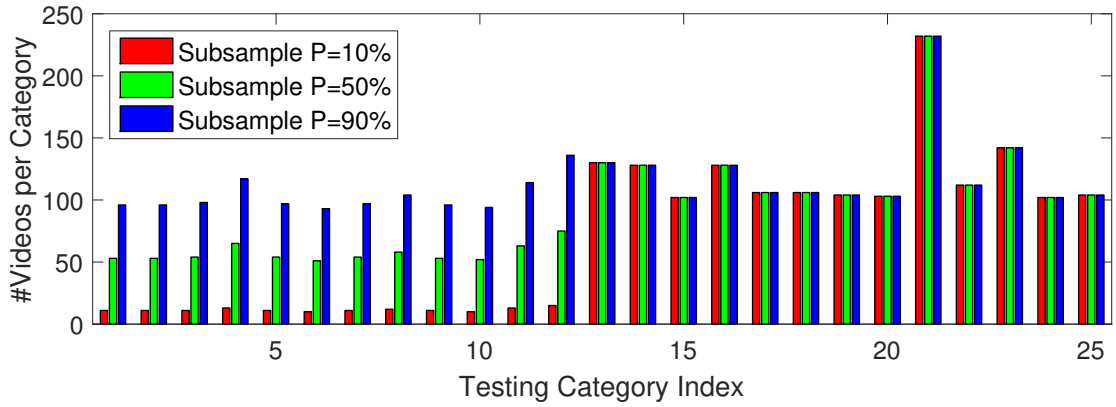


Figure 5.8: Distribution of testing videos after subsampling.

Then we experiment the baseline model - NN and two transductive variants - NN+ST and NN+NRM. By increasing P from 10 to 90 we observe from Figure 5.9 that both self-training (red) and hubness correction (green) improve consistently over non-transductive baseline (black dashed). This suggests our transductive strategies are robust to unbalanced test set.

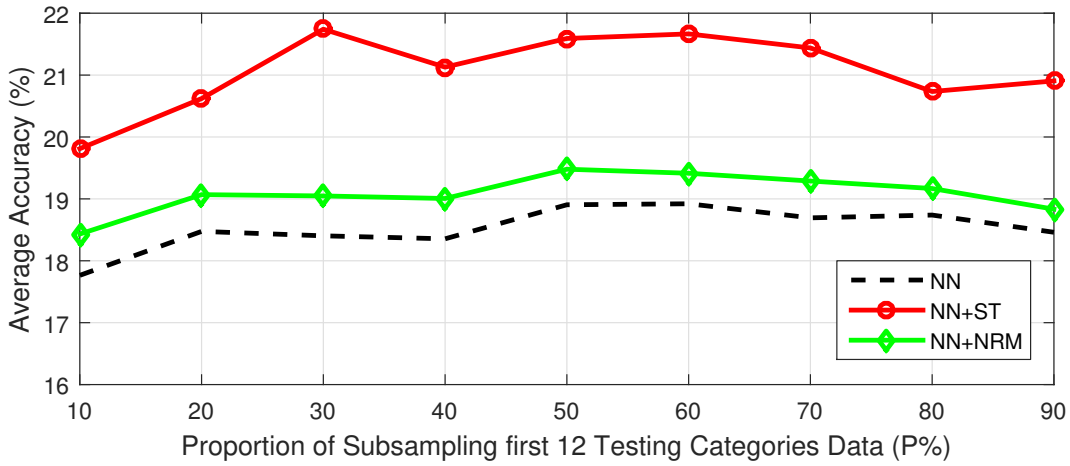


Figure 5.9: Performance of ZSL for subsampled imbalanced test set.

5.3.7 Multi-Shot Learning

We have thus far focused on the efficacy of unsupervised word-vector embeddings for zero-shot learning. In this section we verify that the same representation also performs comparably to state-of-the-art for standard supervised (multi-shot) action recognition. We use the standard data splits and evaluation metrics for all 4 datasets.

Alternatives Models

We compare our approach to:

1. **Low-Level Feature [87]** the state-of-the-art results based on using fisher vector encoded improved trajectory feature [15] with linear SVM classifier.

2. **Human-Labelled Attribute (HLA) [195]** Exploits an alternative semantic space using human labelled attributes. The model trains binary linear SVM classifiers for attribute detection and uses the vector of attribute scores as a representation. A SVM classifier with RBF kernel is then trained on attribute representation to predict final labels.

3. **Data Driven Attribute (DDA) [195]** Learns attributes from data using dictionary learning. These attributes are complementary to the human labelled ones. Automatically discovered attributes are processed in the same way as HLA for action recognition.

4. **Mixed attributes (Mix) [195]** A combination of HLA and DDA is applied to exploit the complementary information in two attribute sets.

5. **Semantic embedding model (Embedding)** first learns a word-vector embedding based on regularized linear regression, as in ZSL. But the standard supervised learning data-split is adopted. All data are mapped into the semantic space via regression and a linear SVM classifier is trained for each category with the mapped training data.

The resulting accuracies are shown in Table 5.10. We observe that our semantic embedding is comparable to the state-of-the-art low-level feature-based classification and is comparable or slightly better than the conventional attribute-based intermediate representations despite the fact that no supervised manual attribute definition and annotation is required.

Table 5.10: Standard supervised action recognition. Average accuracy for HMDB51 and UCF101 datasets. Mean average precision for Olympic Sports and CCV.

Method	HMDB51	UCF101	Olympic Sports	CCV
Low-Level Feature [87]	58.4	84.6	92.1	68.0
HLA [195]	-	81.7	-	-
DDA [195]	-	79.0	-	-
Mix [195]	-	82.3	-	-
Embedding	56.4	82.0	93.4	51.6

5.3.8 Efficiency and Runtime

The efficiency of our ZSL algorithm compares favourably against existing alternatives due to its closed-form solution to mapping the visual feature space to word-vector semantic embedding space (Eq. 5.9). For instance, it took about 300 seconds to train and test 50 splits of the entire HMDB51 benchmark dataset (6766 videos of 51 categories of actions), or 520 seconds with data augmentation, using a single thread on a Intel E5-2680 CPU. The computational cost is dominated by the matrix inversion in Eq. 5.9, which can be sped up by exploiting efficient matrix libraries.

5.3.9 Further Analysis

In the main experiments we set the free parameters ridge regularizor $\gamma_A = 1^{-6}$, manifold regularizor $\gamma_I = 40$, manifold Knn graph $N_K^G = 5$, Self-Training Knn $N_K^{st} = 100$. In this section we analyse the impact of these free parameters in our model.

Word-Vector Dimension

We investigate how the specific word-vector model $\mathbf{z} = g(y)$ affects the performance of our framework. Since Google News Dataset is not publicly accessible we use a smaller but public dataset - 4.6M Wikipedia documents to study the word-vector dimension. In specific, we train word-vectors on 4.6M Wikipedia documents and vary dimension from 32 to 1024. We then evaluate the performance of zero-shot and multishot learning v.s. different dimension of embedding space. The results are given in Figure 5.10.

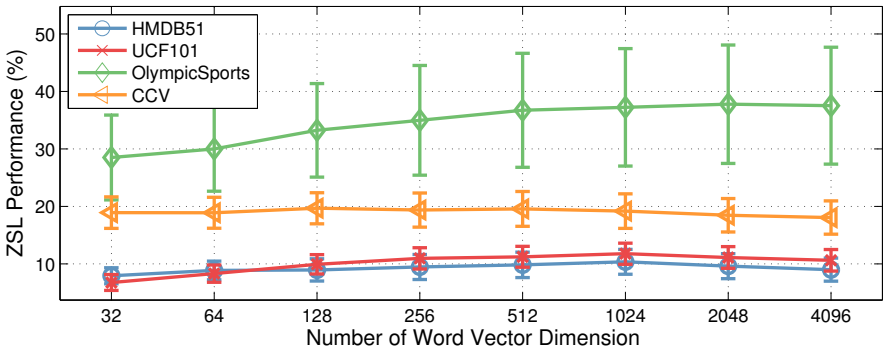


Figure 5.10: Zero-shot performance v.s. dimension of word-vector.

We observe that word-vector dimension does affect the zero-shot recognition performance. Performance generally increases with dimension of word-vector from 32 to 4096 in HMDB51,

UCF101 and Olympic Sports, while showing no clear trend for CCV. In general a reasonable word-vector dimension is between 256 to 2048.

Visual-to-Semantic Mapping

Ridge regression regularization We learn the visual-to-semantic mapping with regularized linear regression. The regularization parameter γ_A controls the regression model complexity. Here, we study the impact of γ_A on zero-shot performance. We measure the 50 splits' average accuracy by varying γ_A in the range of $\{0, 1^{-9}, 1^{-8}, \cdots, 1^{-3}\}$. A plot of zero-shot mean accuracy v.s. regularization parameter is given in Figure 5.11. From this figure we observe that our model is insensitive to the ridge parameter for any non-zero regularizer. However, when no regularization is used the performance is close to random. This is due to all zero or co-linear rows/columns in the kernel matrix which causes numerical problems in computing the inverse.

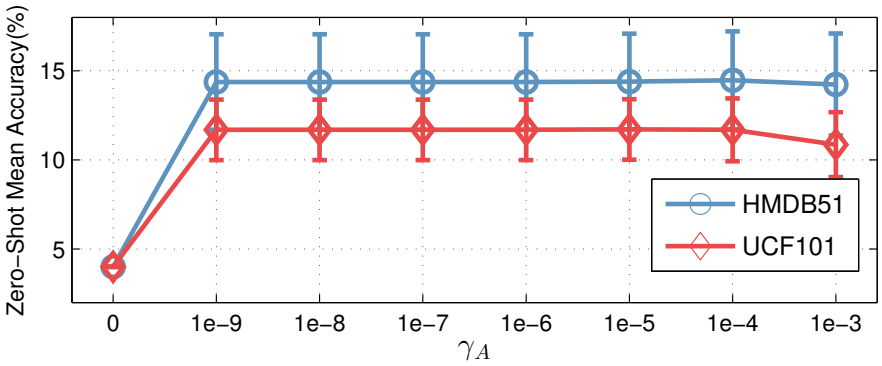
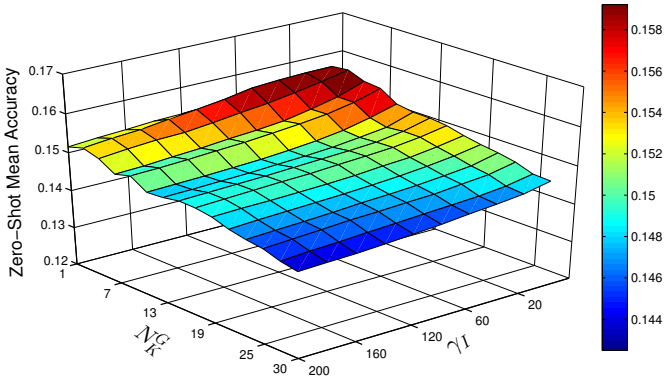


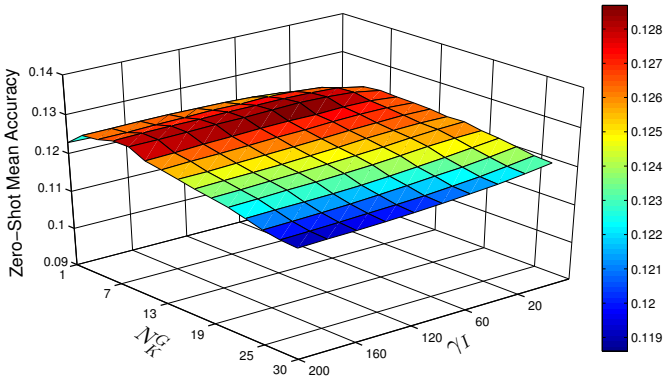
Figure 5.11: Zero-shot mean accuracy v.s. ridge regression parameter

Manifold regression We have seen that transductively exploiting testing/unlabelled data in manifold learning improves zero-shot performance. Two parameters are involved: the manifold regularization parameter γ_I in Loss function (Eq. 5.8) and N_K^G in constructing the symmetrical KNN graph. γ_I controls the preference for preserving the manifold structure in mapping to the semantic space, versus exactly fitting the training data. Parameter N_K^G determines the precision in modelling the manifold structure. Small N_K^G may more precisely exploit the testing data manifold, however it is more prone to noise in the neighbours.

Here we analyse the impact of these two parameters, γ_I and N_K^G by measuring zero-shot recognition accuracy on HMDB51 and UCF101. We evaluate the joint effect of γ_I and N_K^G while fixing $\gamma_A = 1^{-6}$. Specifically we test $\gamma_I \in \{20, 40, \cdots, 100\}$ and $N_K^G \in \{1, 3, 5 \cdots, 29\}$. The results in Figure 5.12 show that there is a slightly preference towards moderately low values of N_K^G and



(a) HMDB51



(b) UCF101

Figure 5.12: Zero-shot recognition accuracy with respect to manifold regression parameters γ_I and N_K^G.

γ_I, but the framework is not very sensitive to these parameters.

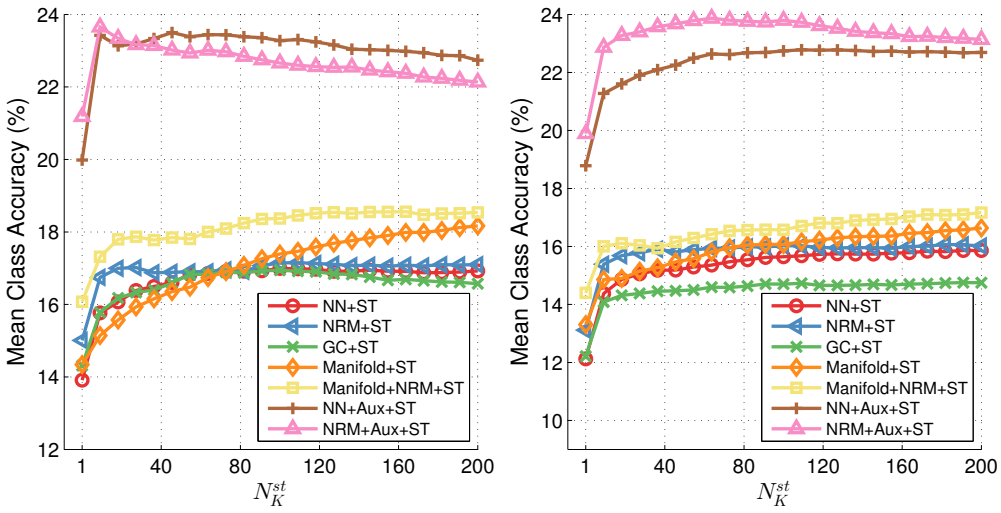
Self-Training

We previously demonstrated in Table 5.4, that self-training (Section 5.2.3) helps to mitigate the domain shift problem. Here, we study the influence of the N_K^{st} parameter for KNN in self-training. Note the N_K^{st} concerns the neighbouring data distribution around prototypes at testing time rather than manifold regularization KNN graph N_K^G at training time. We evaluate $N_K^{st} \in \{1,2,3,\cdots,200\}$. To thoroughly examine the effectiveness of self-training, we investigate all baselines with self-training introduced in Section 5.3.2 including

- X-RR-✓-NN-✓ (**NN+ST**)
- X-RR-✓-NRM-✓ (**NRM+ST**)
- X-RR-✓-GC-✓ (**GC+ST**)
- X-MR-✓-NN-✓ (**Manifold+ST**)
- X-MR-✓-NRM-✓ (**Manifold+NRM+ST**)

- X-MR-✓-NRM-✓ (**Manifold+NRM+ST**)
- ✓-RR-✓-NN-✓ (**NN+Aux+ST**)
- ✓-RR-✓-NRM-✓ (**NRM+Aux+ST**)

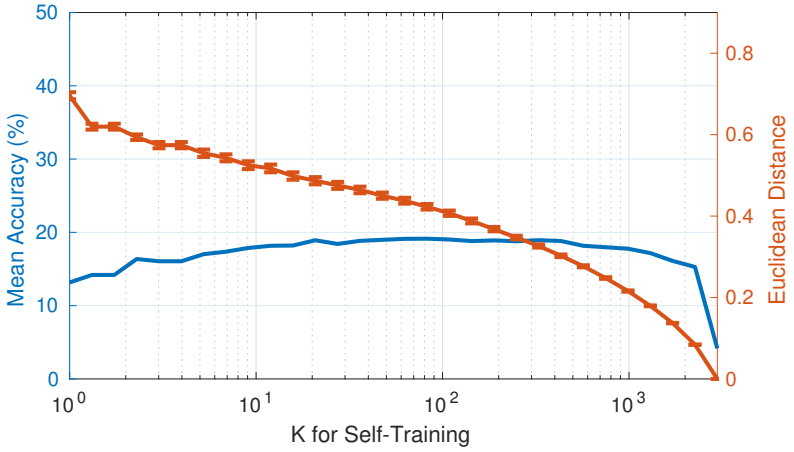
The accuracy v.s. N_K^{st} is illustrated in Figure 5.13. Performance is robust to N_K^{st} when N_K^{st} is above 20.



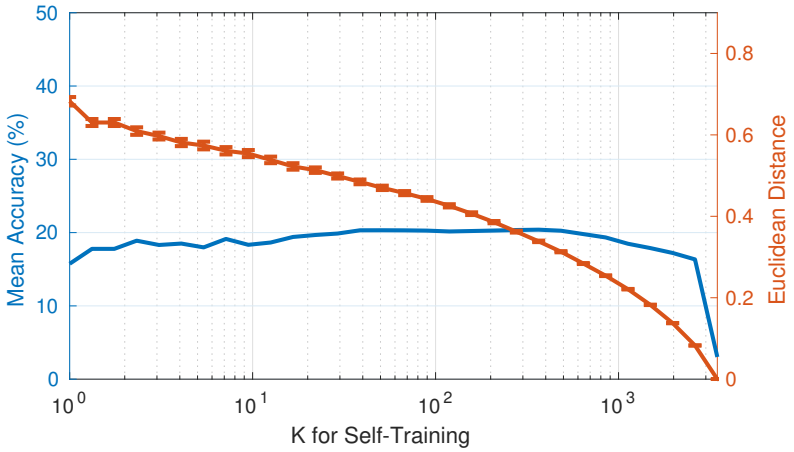
(a) HMDB51 (b) UCF101

Figure 5.13: Zero-shot recognition accuracy v.s. self-training parameter K.

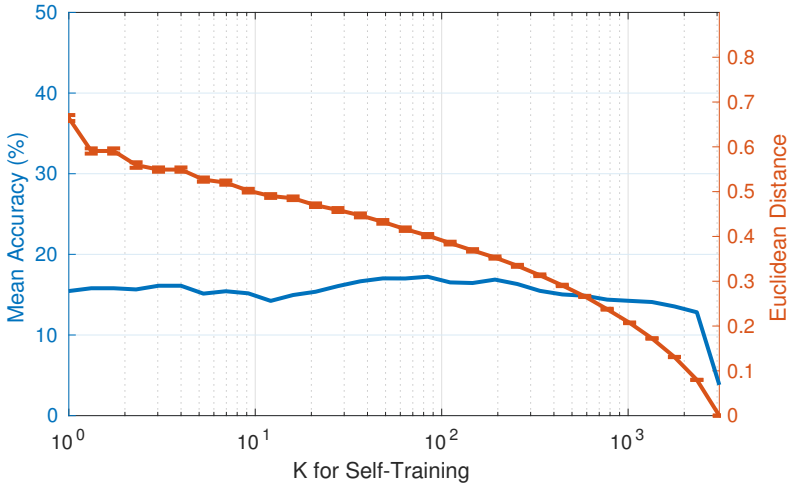
As we note, if the parameter K of self-training is set to the number of testing data all testing category prototypes would converge to a single point. To examine the effect of K parameter w.r.t. the adjusted prototypes we conduct an ad hoc experiment on HMDB51 dataset. In specific, we vary the parameter K for Self-Training from K = 1 to the number of all testing data. We illustrate the testing mean accuracy as blue and average & deviation of the distances of all testing prototypes to the centre of all testing data due to Self-Training as red in Figure 5.14. Three individual random splits are evaluated here. These illustrations suggest that the distance to centre constantly decrease by increasing the Knn of testing prototypes. So the prototypes are moving gradually towards the centre of data distribution. Moreover, The recognition accuracy therefore approaches random guess (4%).



(a) Random Split 1



(b) Random Split 2



(c) Random Split 3

Figure 5.14: Evaluation of Self-Training K parameter v.s. testing accuracy & distance of adjusted prototypes to the centre of all testing data.

5.4 Summary

In this chapter, we investigated *unsupervised* word-vector semantic representation for zero-shot action recognition. The fundamental challenge of zero-shot learning is the disjoint training and testing classes, and associated domain-shift. We explored the impact of four simple but effective strategies to address this: data augmentation, manifold regularization, self-training and hubness correction. Overall we demonstrated that given transductive access to testing data during training stage these strategies are complementary, and together facilitate a highly effective system that outperforms significantly existing methods for zero-shot recognition despite their use of strongly *supervised* embeddings (attributes). Moreover, our model has a closed-form and is very simple to implement (a few lines of matlab) and very efficient to run compared to existing state-of-the-art ZSL methods. Finally, we also provide a unique analysis of the inter-class affinity for ZSL, giving insight into why and when ZSL works. This provides for the first time two new capabilities: the ability to predict the efficacy of a given ZSL scenario in advance, and a mechanism to guide the construction of suitable training sets for a desired set of target classes.

Nevertheless, direct nearest neighbour matching in high-dimension word-vector semantic space is not the optimal solution as this could result in poor distance measure [60]. Furthermore, naive data augmentation does not guarantee all training data are contributing positively to the recognition of novel categories. To deal with both issues, in the next chapter we discover a latent space from original semantic representation in which nearest neighbour matching is more meaningful. Besides, we propose to weight auxiliary data to selectively augment data for better zero-shot recognition. Both approaches can be seamlessly integrate with the current zero-shot framework.

Chapter 6

Multi-Task Semantic Embedding with Prioritised Data Augmentation

---

The preceding chapter discusses an approach to exploiting semantic word-vector space for zero-shot human action recognition. The proposed pipeline aims at establishing a mapping connecting low-level features and a semantic description of the label space, referred to as visual-semantic mapping, on auxiliary training data. Re-using the learned mapping to project target testing videos into an embedding space thus allows novel-classes to be recognised by nearest neighbour inference. However, the proposed zero-shot learning (ZSL) methods suffer from auxiliary-target domain shift intrinsically induced by assuming the same mapping for the disjoint auxiliary and target classes. This compromises the generalisation accuracy of ZSL recognition on the target data. In this chapter, we improve the ability of ZSL to generalise across this domain shift in both model- and data-centric ways by formulating a visual-semantic mapping with better generalisation properties and a dynamic data re-weighting method to prioritise auxiliary data that are relevant to the target classes. An illustration of the proposed approaches is presented in Figure 6.1. Specially: (1) we introduce a multi-task visual-semantic mapping to improve generalisation by constraining the semantic mapping parameters to lie on a low-dimensional manifold; and (2) we explore prioritised data augmentation by expanding the pool of auxiliary data with additional instances weighted by relevance to the target domain. The proposed new model is applied to the challenging zero-shot action recognition problem to demonstrate its advantages over existing ZSL models.

In the remainder of this chapter, we first introduce the multi-task visual-semantic mapping in Section 6.1. In the following Section 6.2, we introduce a training data re-weighting strategy to account for the distribution of unlabelled testing data and categories. Then we present experiments on 3 human action datasets and compare with state-of-the-art models in Section 6.3.

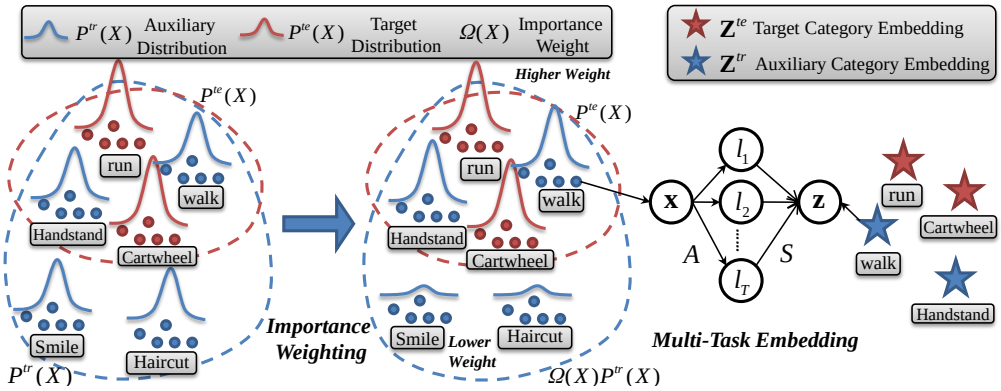


Figure 6.1: Two strategies to improve generalisation of visual-semantic mapping in ZSL. Left: Importance weighting to prioritise auxiliary data relevant to the target domain. Right: Learning the mapping from visual features X to semantic embedding Z by MTL reduces overfitting, and also provides a latent lower dimensional representation $\{l_t\}$ to benefit nearest neighbour matching.

6.1 Visual-Semantic Mapping via Multi-Task Regression

In this chapter, we generally follow the notation defined in the previous section. In ZSL, we aim to recognise action categories $\mathbf{y}$ given visual features $\mathbf{X}$ where training/auxiliary and testing/target categories do not overlap $\mathbf{y}^{tr} \cap \mathbf{y}^{te} = \emptyset$. The key method by which ZSL is achieved is to embed each category label in $\mathbf{y}$ into a semantic label embedding space $\mathbf{Z}$ which provide a vector representation of any *nameable* category.

6.1.1 Training a Visual-to-Semantic Mapping

We first introduce briefly the conventional single task learning using regression for visual-semantic mapping [46, 22].

Single-Task Regression

Following the same word-vector semantic embedding described in Section 5.1.1, we can obtain the class name embedding for a video clip as $\mathbf{z}_i = g(y_i)$. We then learn a visual-semantic mapping function $f : \mathbf{X} \rightarrow \mathbf{Z}$ on the training categories. Given a loss function $l(\cdot,\cdot)$, we learn the mapping f by optimising Eq (6.1) where $\Omega(f)$ denotes regularization on the mapping:

$$\min_{f} \frac{1}{n_l} \sum_{i=1}^{n_l} l\left(f(\mathbf{x}_i), \mathbf{z}_i\right) + \Omega(f). \tag{6.1}$$

The most straightforward choice of mapping f and loss l is linear $f(\mathbf{x}) = \mathbf{W}\mathbf{x}$, and square error respectively, which results in a regularized linear (ridge) regression problem: $l\left(f(\mathbf{x}_i), \mathbf{z}_i\right) = ||\mathbf{z}_i - \mathbf{W}\mathbf{x}_i||_2^2$. A closed-form solution to $\mathbf{W}$ can then be obtained by $\mathbf{W} = \mathbf{Z}\mathbf{X}^\top \left(\mathbf{X}\mathbf{X}^\top + \lambda n_l \mathbf{I}\right)^{-1}$. Each row $\mathbf{w}_d$ of regressor $\mathbf{W}$ maps visual feature $\mathbf{x}_i$ to dth dimension of response variable $\mathbf{z}_i$. Since regressors $\{\mathbf{w}_d\}_{d=1\cdots d_z}$ are learned independently from each other this is referred to as **single-task learning (STL)** with each $\mathbf{w}_d$ defining one distinct 'task'.

From Single to Multi-Task Regression

In the conventional ridge-regression solution to Eq. (6.1), each task $\mathbf{w}_d$ is effectively learned separately, ignoring any relationship between tasks. We wish to model this relationship by discovering a latent basis of predictors such that tasks $\mathbf{w}_d$ are constructed as linear combinations of T latent tasks $\{\mathbf{a}_t\}_{t=1\cdots T}$. So the dth regression predictor is now modelled as $\mathbf{w}_d = \sum_t s_{dt}\mathbf{a}_t = \mathbf{s}_d^\top \mathbf{A}$, where $\mathbf{s}_d$ is the combination coefficient for d-th task. Denoting multi-task regression prediction as $f(\mathbf{x}_i, \mathbf{S}, \mathbf{A})$, we now optimise:

$$\min_{\mathbf{S},\mathbf{A}} \tfrac{1}{n_l} \sum_{i=1}^{n_l} l(f(\mathbf{x}_i, \mathbf{S}, \mathbf{A}), \mathbf{z}_i) + \lambda\Omega(\mathbf{S}) + \gamma\Psi(\mathbf{A}). \tag{6.2}$$

Grouping and Overlap Multi-Task Learning (GOMTL)

An effective method following the MTL design pattern above is GOMTL [62]. GOMTL uses a $\mathbf{W} = \mathbf{S}\mathbf{A}$ task parameter matrix factorisation, where the number of latent tasks T (typically $T < d_z$) is a free parameter. Requiring the combination coefficients $\mathbf{s}_t$ to be sparse, via a ℓ_1 regulariser, the loss is written as

$$\min_{\{\mathbf{s}_t\},\mathbf{A}} \sum_{t=1}^{T} \frac{1}{n_l} \sum_{i=1}^{n_l} ||\mathbf{z}_{t,i} - \mathbf{s}_t \mathbf{A}\mathbf{x}_i|| + \lambda \sum_{t=1}^{T} ||\mathbf{s}_t||_1 + \gamma ||\mathbf{A}||_F^2 \tag{6.3}$$

This can be solved by iteratively updating $\mathbf{A}$ and $\mathbf{S}$. When $\mathbf{A}$ is fixed, loss function reduces to a standard L1 regularized (LASSO) regression problem that can be efficiently solved by Alternating Direction Method of Multipliers (ADMM) [196]. When $\mathbf{S}$ is fixed, we can efficiently solve $\mathbf{A}$ by gradient descent.

Regularized Multi-Task Learning (RMTL)

The classic RMTL method [61] models task parameters as the sum of a globally shared and task specific parameter vector: $\mathbf{w}_t = \mathbf{a}_0 + \mathbf{a}_t$. It can be seen that this corresponds to a special case of GOMTL's $\mathbf{W} = \mathbf{SA}$ predictor matrix factorisation [142]. Here there are $T = d_z + 1$ latent tasks, a fixed task combination vector $\mathbf{s}_t = [1 \quad \mathbb{1}(t=1) \quad \mathbb{1}(t=2) \cdots \mathbb{1}(t=d_z)]^\top$ where $\mathbb{1}(\cdot)$ is the indicator function and $A = \left[\mathbf{a}_0^\top \mathbf{a}_1^\top \cdots \mathbf{a}_{d_z}^\top\right]^\top$.

Explicit Multi-Task Embedding (MTE)

In GOMTL Eq (6.3), it can be seen that the label embedding $\mathbf{z}_i$ is approximated from the data by the mapping $\mathbf{s}_t\mathbf{A}\mathbf{x}_i$, and this approximation is reached by combination via the latent representation $\mathbf{A}\mathbf{x}_i$. While GOMTL defines this space implicitly via the learned $\mathbf{A}$, we propose to model it explicitly as $\mathbf{l}_i \approx \mathbf{A}\mathbf{x}_i$. This is so the actual projections $\mathbf{l}_i$ in this latent space can be regularised explicitly, in order to learn a latent space which generalises better to test data, and hence improves ZSL matching later.

Specifically, we split the GOMTL loss $||\mathbf{z}_i - \mathbf{SA}\mathbf{x}_i||_2^2$ into two parts: $||\mathbf{l}_i - \mathbf{A}\mathbf{x}_i||_2^2$ and $||\mathbf{z}_i - \mathbf{S}\mathbf{l}_i||_2^2$ to learn the mapping to the latent space, and from the latent space to the label embedding respectively. This allows us to place additional regularization on $\mathbf{l}_i$ to avoid extreme values in the latent space and thus later improve neighbour matching (Section 6.1.2). Given the large and high dimensional video datasets, we apply Frobenius norm on $\mathbf{S}$ in contrast to GOMTL's ℓ_1.

$$\begin{aligned}\min_{\{\mathbf{s}_t\},\mathbf{A},\{\mathbf{l}_i\}} \quad & \sum_{t=1}^{T} \frac{1}{n_l} \sum_{i=1}^{n_l} \left(||\mathbf{z}_{t,i} - \mathbf{s}_t\mathbf{l}_i||_2^2 + ||\mathbf{l}_i - \mathbf{A}\mathbf{x}_i||_2^2 \right) + \\ & \lambda_S \sum_{t=1}^{T} ||\mathbf{s}_t||_2^2 + \lambda_A ||\mathbf{A}||_F^2 + \lambda_L \sum_{i=1}^{n_l} ||\mathbf{l}_i||_2^2\end{aligned} \tag{6.4}$$

Our explicit multi-task embedding has similarities to [23], but our purpose is multi-task regression for ZSL, rather than embedding for video descriptions. To solve our explicit embedding model we iteratively solve $\mathbf{L}$,$\mathbf{A}$ and $\mathbf{S}$ while fixing the other two. With the ℓ_2 norm on $\mathbf{S}$, this has a convenient closed-form solution to each parameter:

$$\begin{aligned}\mathbf{L} &= (\mathbf{S}^\top\mathbf{S} + (\lambda_L n_l + 1)\mathbf{I})^{-1}(\mathbf{S}^\top\mathbf{Z} + \mathbf{A}\mathbf{X}) \\ \mathbf{S} &= \mathbf{Z}\mathbf{L}^\top(\mathbf{L}\mathbf{L}^\top + \lambda_S n_l \mathbf{I})^{-1} \\ \mathbf{A} &= \mathbf{L}\mathbf{X}^\top(\mathbf{X}\mathbf{X}^\top + \lambda_A n_l \mathbf{I})^{-1}\end{aligned} \tag{6.5}$$

6.1.2 Zero-Shot Action Recognition

We consider two alternative NN matching methods for zero-shot action prediction that use the MTL mappings described above.

Distributed Space Matching

Given a trained visual-to-semantic regression f, we project testing set visual feature $\mathbf{x}^{te}$ into the semantic label embedding space. The standard strategy [22, 46] is then to employ NN matching in this space for zero-shot recognition. Specifically, given the class name embedding function $g(\cdot)$, and using cosine distance norm, the testing video $\mathbf{x}^{te}$ are classified by:

$$y^* = arg \min_{y^* \in \mathbf{y}^{te}} ||g(y^*) - f(\mathbf{x}^{te})|| \tag{6.6}$$

where $f(\mathbf{x}^{te}) = \mathbf{W}\mathbf{x}^{te}$ for STL and $f(\mathbf{x}^{te}) = \mathbf{S}\mathbf{A}\mathbf{x}^{te}$ for MTL.

Latent Space Matching

MTL methods provide an alternative to matching in label space: Matching in the latent space. The representation of testing data in this space is the output of latent regressors $\mathbf{l}^{te} = \mathbf{A}\mathbf{x}^{te}$ (Eq. (6.4)). To get the representation of testing categories in the latent space we invert the combination matrix $\mathbf{S}$ to project target category name embeddings $g(\mathbf{y}^{\mathbf{te}})$ into latent space. In specific we classify by Eq. (6.7), where $(\mathbf{S}^\top\mathbf{S})^{-1}\mathbf{S}^\top$ is the Moore-Penrose pseudoinverse.

$$\mathbf{y}^* = arg \min_{y^* \in \mathbf{y}^{te}} ||(\mathbf{S}^\top\mathbf{S})^{-1}\mathbf{S}^\top g(y^*) - \mathbf{A}\mathbf{x}^{te}|| \tag{6.7}$$

NN matching in the latent space is better than in semantic label space because: (1) the dimension is lower $T < d_z$, and (2) we have explicitly regularised the latent space to be well behaved (Eq. (6.4)).

6.2 Importance Weighting

Augmenting auxiliary data with additional examples from other datasets has been proved to benefit learning the visual-semantic mapping in the previous chapter. However, simply aggregating auxiliary and additional datasets is not ideal as including irrelevant data risks 'negative transfer'. Therefore we are motivated to develop methodology to prioritise augmented training data that is useful for a particular ZSL recognition scenario. Specifically, we learn a per-instance weighting

$\omega(\mathbf{x})$ on the augmented auxiliary training dataset $\mathbf{X}^{tr}$ to adjust each instance's contribution according to relevance to the target domain. Because Importance Weighting (IW) adapts auxiliary data to the target domain, we assume a transductive setting with access to testing data $\mathbf{X}^{te}$.

6.2.1 Kullback-Leibler Importance Estimation Procedure (KLIEP)

We first introduce the way to estimate a per-instance auxiliary-data weight given the distribution of target data $\mathbf{X}^{te}$. This is based on the idea [58] of minimizing the KL-divergence ($\mathcal{KLD}$) between training $p^{tr}(\mathbf{x})$ and testing data distribution $p^{te}(\mathbf{x})$ via learning a weighting function $\omega(\mathbf{x})$. This is formalised in Eq. (6.8):

$$\begin{aligned} &\min_{\omega} \mathcal{KLD}(p^{te}(\mathbf{x})||\omega(\mathbf{x})p^{tr}(\mathbf{x})) = \int p^{te}(\mathbf{x}) \log \frac{p^{te}(\mathbf{x})}{\omega(\mathbf{x})p^{tr}(\mathbf{x})} d\mathbf{x} \\ &\min_{\omega} \int p^{te}(\mathbf{x}) \log \frac{p^{te}(\mathbf{x})}{p^{tr}(\mathbf{x})} d\mathbf{x} - \int p^{te}(\mathbf{x}) \log \omega(\mathbf{x}) d\mathbf{x} \end{aligned} \tag{6.8}$$

The first term is fixed w.r.t. $\omega(\mathbf{x})$ so the objective to optimise is:

$$\min_{\omega} - \int p^{te}(\mathbf{x}) \log \omega(\mathbf{x}) d\mathbf{x} \approx -\frac{1}{n_x^{te}} \sum_{i=1}^{n_x^{te}} \log \omega(\mathbf{x}_i) \tag{6.9}$$

6.2.2 Aligning Both Visual Features and Labels

KLIEP is conventionally used for domain adaptation by reweighting instances [58, 153]. In the case of transductive ZSL, we have the target data $\mathbf{X}^{te}$ and category labels $\mathbf{Z}^{te}$ respectively, although not instance-label association which is to be predicted. In this case we can further improve ZSL by extending KLIEP to align training and testing sets in both visual feature and category sense. Though KLEIP with labels was studied by Garcke et al.[59], but they assumed the target joint distribution of $\mathbf{X}$ and $\mathbf{Z}$ is known. So Garcke et al.[59] is only suitable for traditional supervised learning with labelled target examples of $\mathbf{z}_i$ and $\mathbf{x}_i$ in correspondence. In our case we have the videos to classify and the zero-shot category names, but the assignment of names to videos is our task rather than prior knowledge. Specifically, we minimise the kullback-leibler divergence between the target and auxiliary in terms of both the visual and category distributions:

$$\begin{aligned} &\min_{\omega_x,\omega_z} \mathcal{KLD}(p^{te}(X)||\omega_x(\mathbf{X})p^{tr}(\mathbf{X})) + \mathcal{KLD}(p^{te}(\mathbf{Z})||\omega_z(\mathbf{Z})p^{tr}(\mathbf{Z})) \\ &\min_{\omega_x,\omega_z} -\frac{1}{n_x^{te}} \sum \log \omega_x(\mathbf{x}_i^{te}) - \frac{1}{n_x^{te}} \sum \log \omega_z(\mathbf{z}_i^{te}) \end{aligned} \tag{6.10}$$

Given both $\mathbf{X}^{te}$ and $\mathbf{Z}^{te}$, we construct the weighting functions as a combination of Gaussian kernels centred at the testing data and categories. Specifically we define $\omega(\mathbf{x},\mathbf{z}) = \omega_x(\mathbf{x}) + \omega_z(\mathbf{z})$ and $\omega_x(\mathbf{x})$ and $\omega_z(\mathbf{z})$ are calculated as in Eq. (6.11) where n_u is the number of unlabelled testing data. Here $\omega(\mathbf{x},\mathbf{z})$ extends the previous notation $\omega(\mathbf{x})$ to indicate giving a weight to each training instance given visual feature $\mathbf{x}$ and class name embedding $\mathbf{z}$. So if there are n_l labelled training instances, $\omega(\mathbf{x},\mathbf{z})$ returns a weight vector of the same length.

$$\begin{aligned}
\omega_x(\mathbf{x}) &= \sum_{i=1}^{n_u} \alpha_i \phi(\mathbf{x}, \mathbf{x}_i^{te}), \\
\omega_z(\mathbf{z}) &= \sum_{i=1}^{n_u} \beta_j \phi(\mathbf{z}, \mathbf{z}_i^{te}), \\
\phi(\mathbf{x}, \mathbf{x}_i^{te}) &= exp\left(-\frac{||\mathbf{x} - \mathbf{x}_i^{te}||^2}{2\sigma^2}\right)
\end{aligned} \tag{6.11}$$

For ease of formulation, we denote $\mathbf{a} = [\alpha_1 \cdots \alpha_{n_u}]^\top$, $\mathbf{b} = [\beta_1 \cdots \beta_{n_u}]^\top$, $\Phi_\mathbf{a}(\mathbf{x}) = [\phi(\mathbf{x}, \mathbf{x}_1^{te}) \cdots \phi(\mathbf{x}, \mathbf{x}_{n_u}^{te})]^\top$ and $\Phi_\mathbf{b}(\mathbf{z}) = [\phi(\mathbf{z}, \mathbf{z}_1^{te}) \cdots \phi(\mathbf{z}, \mathbf{z}_{n_u}^{te})]^\top$. The optimization can be thus written as

$$\begin{aligned}
&\min_{\mathbf{a},\mathbf{b}} -\frac{1}{n_u}\sum_{i=1}^{n_u} \log \mathbf{a}^\top \Phi_\mathbf{a}(\mathbf{x}_i^{te}) - \frac{1}{n_u}\sum_{i=1}^{n_u} \log \mathbf{b}^\top \Phi_\mathbf{b}(\mathbf{z}_i^{te}), \\
&s.t. \frac{1}{n_l}\sum_{i=1}^{n_l} \omega(\mathbf{x}_i^{tr}, \mathbf{z}_i^{tr}) = 1
\end{aligned} \tag{6.12}$$

The above constrained optimization problem is convex w.r.t. both $\mathbf{a}$ and $\mathbf{b}$. It can be solved by interior point methods using the derivatives in Eq. (6.13):

$$\begin{aligned}
\nabla \mathbf{a} &= -\frac{1}{n_u}\sum_{i=1}^{n_u} \frac{1}{\mathbf{a}^\top \Phi_\mathbf{a}(\mathbf{x}_i^{te})} \Phi_\mathbf{a}(\mathbf{x}_i^{te}), \\
\nabla \mathbf{b} &= -\frac{1}{n_u}\sum_{i=1}^{n_u} \frac{1}{\mathbf{b}^\top \Phi_\mathbf{b}(\mathbf{z}_i^{te})} \Phi_\mathbf{b}(\mathbf{z}_i^{te})
\end{aligned} \tag{6.13}$$

6.2.3 Weighted Visual-to-Semantic Regression

Given per-instance weights ω estimated above, we can rewrite the loss function for both single-task ridge regression and multi-task regression in Sec 6.1.1 as $\omega_i l(f(\mathbf{x}_i, \mathbf{A}), \mathbf{z}_i)$ and $\omega_i l(f(\mathbf{x}_i, \mathbf{S}, \mathbf{A}), \mathbf{z}_i)$ respectively. All our loss functions have quadratic form, so the weight can be expressed inside the quadratic loss e.g. $\omega_i||\mathbf{z}_i - \mathbf{W}\mathbf{x}_i||_2^2 = ||\mathbf{z}_i\sqrt{\omega_i} - \mathbf{W}\mathbf{x}_i\sqrt{\omega_i}||_2^2$. Thus to incorporate the weight information we simply replace the original semantic embedding matrix with $\tilde{\mathbf{z}}_i = \mathbf{z}_i\sqrt{\omega_i}$ and data matrix with $\tilde{\mathbf{x}}_i = \mathbf{x}_i\sqrt{\omega_i}$.

6.2.4 Relation to Other Learning Strategies

There is a distinctive difference between our importance weighting strategy and active learning. Active Learning aims to select a subset of unlabelled data for user to label. As a result, the model updated with new labelled data is expected to produce better performance. In contrast, our importance weighting model make no assumption of additional user annotation but instead selectively use related labelled data. Therefore, importance weighting is more closely related to domain adaptation.

The classical AdaBoost adjust the weight for each training examples by giving higer weight to instance incorrectly classified or with large error for regression [197]. The more recent transfer boosting models [153] extends the AdaBoost to the scenario where there are one target and multiple source domains. All of these boosting models assume the target domain are labelled. Nonetheless we dont make this assumption in our importance weighting model and our target is to classify the target data into novel categories.

6.3 Experiments

6.3.1 Datasets and Settings

We evaluated our proposed approaches on three human action recognition datasets, HMDB51 [4], UCF101 [5] and Olympic Sports [7]. We use the same fisher vector encoded improved trajectory feature introduced in the previous chapter. They contain 6766, 13320, 783 videos and 51, 101, 16 categories respectively. For all datasets we extract improved trajectory feature (ITF) [15]. We use Fisher Vectors (FV) [92] to encode three raw descriptors (HOG, HOF and MBH). Each descriptor is reduced to half of its original dimension by PCA, resulting in a 198 dim representation. Then we randomly sample 256,000 descriptors from all videos and learn a Gaussian Mixture with 128 components to obtain the FVs. The final dimension of FV encoded feature is $2 \times 128 \times 198 = 50688$ dimensions. For the label-embedding, we use 300-dimensional word2vec [24]. We use $T = n_c^{tr}$ latent tasks, and cross-validation to determine regularisation strength hyper-parameters for the models. It is worth noting that given the video data and visual feature we extracted, Ridge Regression (RR) has 15M (300×50688) parameters, whilst for HMDB51 where $T = 25$, GOMTL and MTE have 1.27M ($50688 \times 25 + 25 \times 300$) parameters.

Table 6.1: Visual-to-semantic mappings for zero-shot action recognition: MTL (✓) versus STL (X). Latent matching (✓) versus distributed (X) matching.

ZSL Model	MTL	Latent Matching	HMDB51	UCF101	Olympic Sports
RR	X	NA	18.3 ± 2.1	14.5 ± 0.9	40.9 ± 10.1
RMTL [61]	✓	X	18.5 ± 2.1	14.6 ± 1.1	41.1 ± 10.0
RMTL [61]	✓	✓	18.7 ± 1.7	14.7 ± 1.0	41.1 ± 10.0
GOMTL [62]	✓	X	18.5 ± 2.2	13.1 ± 1.5	43.5 ± 8.8
GOMTL [62]	✓	✓	18.9 ± 1.0	14.9 ± 1.5	44.5 ± 8.5
MTE	✓	X	18.7 ± 2.2	14.2 ± 1.3	$\mathbf{44.5 \pm 8.2}$
MTE	✓	✓	$\mathbf{19.7 \pm 1.6}$	$\mathbf{15.8 \pm 1.3}$	44.3 ± 8.1

6.3.2 Visual-Semantic Mappings for Zero-Shot Action Recognition

Evaluation Criteria

To evaluate zero-shot action recognition, we divide each dataset evenly into training and testing parts with 5 random splits. Using classification accuracy for HMDB51 and UCF101 and average precision for Olympic Sports as the evaluation metric, the average and standard deviation over the 5 splits are reported for each dataset.

Compared Methods

We study the efficacy of our contributions by evaluating the different visual-semantic mappings presented in Sec 6.1.1. We compare MTL-regression methods with conventional STL Ridge Regression (denoted **RR**) for ZSL. For RR/STL, nearest neighbour matching is used to recognise target categories. Note that the RR+NN method here corresponds to the core strategy used by the previous chapter. The multi-task models we explore include: **RMTL** [61]: assumes each task's predictor is the sum of a global latent vector and a task-specific vector. **GOMTL** [62]: Uses a predictor-matrix factorisation assumption in which tasks' predictors lie on a low-dimensional subspace. **Multi-Task Embedding (MTE)**: Our model differs from GOMTL in that it explicitly models and regularises a lower dimensional latent space. For the multi-task methods, we also compare the ZSL matching strategies introduced in Section 6.1.2: **Distributed:** Standard NN matching (Eq. (6.6)), and **Latent:** our proposed latent-space matching (Eq. (6.7)).

Results

The comparison of single task ridge regression with our multi-task methods is presented in Table 6.1. From these results we make the following observations: (1) overall our multi-task methods improve on the corresponding single-task baseline of RR. MTL regression (RMTL, GOMTL and MTE) improves single-task ridge regression by $5-10\%$ in relative terms, with the biggest margins visible on the Olympic Sports dataset; (2) within multi-task models, the GOMTL with sparse ℓ_1 regularization outperforms RMTL. This suggests learning the task combination $\mathbf{S}$ from data is better than fixing it as in RMTL; (3) our MTE generally outperforms other multi-task methods supporting the explicit modelling and regularisation of the latent space; and (4) in most cases, NN matching in the latent space improve zero-shot performance. This is likely due to the lower dimension of the latent space compared to the dimension of the original word vector embedding, making NN matching more meaningful [60].

6.3.3 Importance Weighted Data Augmentation

We next evaluate the impact of importance weighting in data augmentation for zero-shot action recognition. We perform the same 5 random split benchmark for each dataset. For data augmentation, we augment each dataset's training split with the data from all other datasets. For instance, for ZSL on HMDB51 we augment the training data with all videos from UCF101 and Olympic Sports.

Compared Methods

We study the impact of the data augmentation methods: **Naive DA:** Naive Data Augmentation, adopted in the previous chapter, simply assigns equal weight to each auxiliary training sample. **Visual KLIEP:** The auxiliary data is aligned with the testing sample distribution $\mathbf{X}^{te}$ (Eq. (6.8)). **Category KLIEP:** The auxiliary categories are aligned with testing category distribution $\mathbf{Z}^{te}$. This is achieved by the same prodcedure in Eq. (6.8) by replacing $\mathbf{x}$ with $\mathbf{z}$. **Full KLIEP:** The distribution of both samples $\mathbf{X}^{te}$ and categories $\mathbf{Z}^{te}$ is used to reweight the auxiliary data (Eq. (6.12)).

Results

From the results in Table 6.2, we draw the conclusions: (1) both the baseline single task learning (STL) method and our Multi-Task Embedding (MTE) improve with Naive DA (compare unaugmented results in Table 6.1); (2) the Visual, Category, and Full visual+category-based weightings

Table 6.2: Data augmentation and importance weighting for ZSL action recognition. Note the results in this table is not directly comparable to those Table 5.4 and Table 5.3. Moreover, another single-task embedding approach - manifold regularized regression can be considered for evaluation in combine with Naive DA. However this embedding is of transductive nature.

ZSL Model	Weighting Model	HMDB51	UCF101	OlympicSports
RR	Naive DA	21.9 ± 2.4	19.4 ± 1.7	46.5 ± 9.4
MTE	Naive DA	$\mathbf{23.4 \pm 3.4}$	$\mathbf{20.9 \pm 1.5}$	$\mathbf{49.4 \pm 8.8}$
RR	Visual KLIEP	23.2 ± 2.7	20.3 ± 1.6	47.2 ± 9.3
RR	Category KLIEP	23.0 ± 2.1	20.2 ± 1.6	51.8 ± 8.7
RR	Full KLIEP	23.7 ± 2.7	20.7 ± 1.4	51.3 ± 9.0
MTE	Visual KLIEP	23.4 ± 2.8	20.8 ± 2.0	51.4 ± 9.2
MTE	Category KLIEP	23.3 ± 2.4	20.9 ± 1.7	50.9 ± 8.3
MTE	Full KLIEP	$\mathbf{23.9 \pm 3.0}$	$\mathbf{21.9 \pm 2.7}$	$\mathbf{52.3 \pm 8.1}$

all improve on Naive DA in the case of STL RR; (3) we see that our MTE with Full KLIEP augmentation performs the best overall. The ability of KLIEP to improve on Naive DA suggests that the auxiliary data is indeed of variable relevance to the target data, and selectively re-weighing the auxiliary data is important; and (4) for KLIEP-based DA, either Visual or Category DA provides most of the improvement, with relatively less improvement obtained by using both together.

Alternative Models

We also compare against previous state-of-the-art methods including those driven by both attributes and word-vector category embeddings. **DAP/IAP** [19]: Direct/Indirect attribute prediction are classic attribute-based zero-shot recognition models based on training SVM classifiers independently for each attribute, and using a probabilistic model to match attribute predictions with target classes. **HAA**: We implement a simplified version of the Human Actions by Attributes model [20]: We first train attribute detection SVMs, and test samples are assigned to categories based on cosine distance between their vector of attribute predictions and the target classes' attribute vectors. **SVE** [190]: Support vector regression was adopted to learn the visual-to-semantic mapping. **ESZSL** [45]: Embarrassingly Simple Zero-Shot Learning defines the loss function as the mean square error on label prediction in contrast to the regression loss defined in other baseline models. **SJE**: Structured Joint Embedding [29] employed a triplet hinge loss. The objective is to enforce relevant labels having higher projection values from visual features than

those of non-relevant labels. **UDA**: The Unsupervised Domain Adaptation model [145] learns dictionary on auxiliary data and adapts it to the target data as a constraint on the target dictionary rather than blindly using the same dictionary. This work combines both attribute and word vector embeddings.

Comparison Versus the State-of-the-Art

Table 6.3 compares our models with various contemporary and state-of-the-art models. For clear comparison, we indicate for each method which embedding ((**W**)ordvector / (**A**)ttribute) and feature (our FV, or BoW) are used, as well as whether it has a transductive dependency on the test data (**TD**) or exploits additional augmenting data (**Aug**). From these results we conclude that: (1) although data augmentation has a big impact, our non-transductive and no data augmentation method (MTE) generally outperforms prior alternatives due to learning an effective latent matching space robust to the train/test class shift; (2) the performance of our MTE with word-vector embedding is strong when compared with DAP/IAP/HAA/ESZSL even with attribute embedding. Given the same attribute embedding, MTE outperforms all state-of-the-art models due to the discovery of latent attributes from the original attribute space; (3) moreover, given importance weighting on auxiliary data, our method (MTE + Full KLIEP) with word-vector embedding performs the best overall – including against the full model explored in Chapter 5 which also exploits data augmentation; and (4) finally, our method is synergistic to the transductive post-processing strategies including self-training (ST) and hubness correction strategy (NRM) both of which are introduced in Section 5.2.1. The combined final is termed as (MTE + Full KLIEP + ST + NRM).

6.3.4 Qualitative Results and Further Analysis

Importance Weighting

To visualise the impact of our IW, we randomly select 4 / 16 classes as target / auxiliary sets respectively. We then estimate the weight on the 16 auxiliary video classes according to the Full KLIEP (Section 6.2). Examples of the auxiliary video weightings are presented in Fig 6.2. We observe that auxiliary classes semantically related to the targets are given higher weight e.g. HandstandPushups→Cartwheel in first sample, SalsaSpin→Hug and Sword Exercise → Fencing in the second sample. While the visually and semantically less relevant auxiliary videos are given much lower weights.

Table 6.3: Comparison versus the state-of-the-art. Embed: Label embedding, Feat: Visual feature used, Aug: Data augmentation required? TD: Transductive Requirement? N/A indicates not available due to the absence of attribute annotation or not reported by the original work.

Method	Embed	Feat	TD	Aug	HMDB51	UCF101	Olympic Sports
MTE	W	FV	X	X	19.7 ± 1.6	15.8 ± 1.3	44.3 ± 8.1
MTE + Full KLIEP	W	FV	✓	✓	23.9 ± 3.0	21.9 ± 2.7	52.3 ± 8.1
MTE + Full KLIEP + ST + NRM	W	FV	✓	✓	$\mathbf{24.8 \pm 2.2}$	$\mathbf{22.9 \pm 3.3}$	$\mathbf{56.6 \pm 7.7}$
MTE	A	FV	X	X	N/A	18.3 ± 1.7	55.6 ± 11.3
DAP [19] - CVPR 2009	A	FV	X	X	N/A	15.9 ± 1.2	45.4 ± 12.8
IAP [19] - CVPR 2009	A	FV	X	X	N/A	16.7 ± 1.1	42.3 ± 12.5
HAA [20] - CVPR 2011	A	FV	X	X	N/A	14.9 ± 0.8	46.1 ± 12.4
SVE [190] - ICIP 2015	W	BoW	X	X	14.9 ± 1.8	12.0 ± 1.4	N/A
SVE [190] - ICIP 2015	W	BoW	✓	X	15.6 ± 0.7	16.5 ± 2.4	N/A
SVE [190] - ICIP 2015	W	BoW	X	✓	19.3 ± 4.0	13.1 ± 2.0	N/A
SVE [190] - ICIP 2015	W	BoW	✓	✓	22.8 ± 2.6	18.4 ± 1.4	N/A
ESZSL [45] - ICML 2015	W	FV	X	X	18.5 ± 2.0	15.0 ± 1.3	39.6 ± 9.6
ESZSL [45] - ICML 2015	W	FV	X	✓	22.7 ± 3.5	18.7 ± 1.6	51.4 ± 8.3
ESZSL [45] - ICML 2015	A	FV	X	X	N/A	17.1 ± 1.2	53.9 ± 10.8
SJE [29] - CVPR 2015	W	FV	X	X	13.3 ± 2.4	9.9 ± 1.4	28.6 ± 4.9
SJE [29] - CVPR 2015	A	FV	X	X	N/A	12.0 ± 1.2	47.5 ± 14.8
UDA [145] - ICCV 2015	A	FV	✓	X	N/A	13.2 ± 1.9	N/A
UDA [145] - ICCV 2015	A+W	FV	✓	X	N/A	14.0 ± 1.8	N/A

Multi-task Embedding

We next qualitatively illustrate single versus multi-task visual-semantic mappings. Specifically we take 5 classes to be recognised and visualise their data after visual-semantic projection by tSNE [198]. A comparison between the representations generated by single-task (RR) and multi-task (MTE) mappings is given in Fig 6.3. The multi-task embedding discovers data in a lower dimension latent space where NN classification becomes more meaningful. The improved representation is illustrated by computing the ROC curve for each target category, as seen in Fig 6.3. MTE provides improved detection over RR, demonstrating the better generalisation of this representation.

Figure 6.2: Visualisation of Full KLIEP auxiliary data weighting. Left: 4 target videos with category names. Right: 16 auxiliary videos with bars indicating the estimated weights.

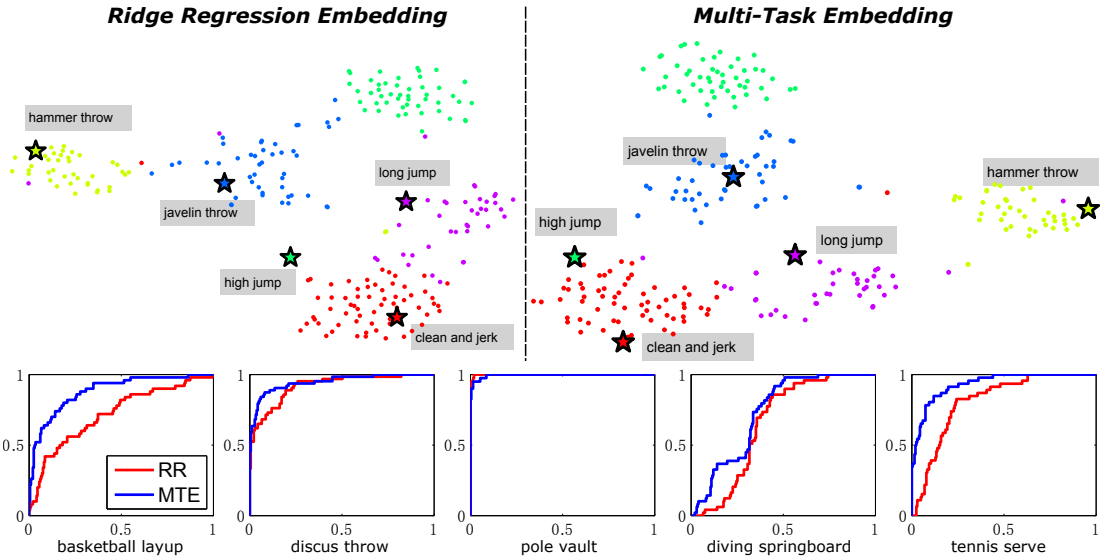


Figure 6.3: Qualitative comparison between single-task ridge regression (RR) and multi-task embedding (MTE).

6.4 Summary

In this chapter, we focus on zero-shot action recognition from the perspective of improving generalisation of the visual-semantic mapping across the disjoint train/test class gap. We propose both model and data-centric improvements to a traditional regression-based pipeline by respectively, multi-task embedding to minimise overfit of the train data and to build a lower dimensional latent matching space; and prioritising data augmentation by importance weighting to best exploit auxiliary data for the recognition of target categories. Our experiments on a set of contemporary action-recognition benchmarks demonstrate the impact of both our contributions and show state-of-the-art results overall.

As discussed, crowd behaviour analysis is another interesting problem to explore apart from

human action analysis due to the potential application in visual surveillance [65]. Crowd behaviours, e.g. street fight and mob flashing, are usually rare in daily surveillance while of more interesting to surveillance operators. To recognise those rare and interesting behaviours without collecting corresponding training examples we explore, in the following chapter, zero-shot crowd behaviour recognition.

Chapter 7

Zero-Shot Crowd Behaviour Analysis

In this chapter we develop a zero-shot multi-label attribute contextual prediction model. We make the assumption that the detection of known attributes helps the recognition of unknown ones. For instance, a putative unknown attribute such as '*violence*' may be related to known attributes '*outdoor*', '*fight*', '*mob*', and '*police*' among others. Therefore, high confidence in these attributes would support the existence of '*violence*'. Specifically, our model first learns a probabilistic P-way classifier on P known attributes, e.g. $p(\text{`}outdoor\text{'}|\mathbf{x})$. Then we estimate the probability of each novel (unseen) attribute conditioned on the confidence of $P$ known attributes, e.g. $p(\text{`}violence\text{'}|\text{`}outdoor\text{'})$. Recall that due to '*violence*' in this example being a novel attribute, this conditional probability cannot be estimated directly by tabulation of annotation statistics. To model this conditional, we consider two contextual learning approaches. The first approach relies on the semantic relatedness between the two attributes. For instance, if '*fight*' is semantically related to '*violence*', then we would assume a high conditional probability $p(\text{`}violence\text{'}|\text{`}fight\text{'})$. Crucially, such semantic relations can be learned in the absence of annotated video data. This is achieved by using large text corpora [143] and language models [24, 25]. However, this text-only based approach has the limitation that linguistic relatedness may not correspond reliably to the visual contextual co-occurence that we wish to exploit. For example, the word '*outdoor*' has high linguistic semantic relatedness, e.g. measured by a cosine similarity, with '*indoor*', whilst they would never co-occur in video annotations. Therefore, our second approach to conditional probability estimation is based on learning to map from *pairwise* linguistic semantic relatedness to visual co-occurence. Specifically, on the known training attributes, we train a bilinear mapping

to map *the pair of* training word-vectors, e.g. $\mathbf{v}(\text{`}fight\text{'})$ and $\mathbf{v}(\text{`}mob\text{'})$, to the training attributes' co-occurrence. This bilinear mapping can then be used to better predict the conditional probability between known and novel/unseen attributes. This is analogous to the standard ZSL idea of learning a visual-semantic mapping from a set of single attributes and re-using this mapping across different unseen attributes. Here, we focus instead on a set of attribute-pairs to learn co-occurrence mapping, and re-using this pairwise mapping across new attribute pairs.

The remainder of this chapter is organised as follows: In Section 7.1, we introduce a general procedure for predicting novel behavioural attributes based on their relation to known attributes. This is formulated as a probabilistic graphic model adapted from Lampert et al.[19] and Gan et al.[199]. We then give the details in Section 7.2 on how to learn a behaviour predictor that estimates the relations between known and novel attributes by inferring from text corpus and co-occurrence statistics of known attribute annotations. Experiments on zero-shot crowd attribute prediction and transfer violence detection are conducted in Section 7.3.

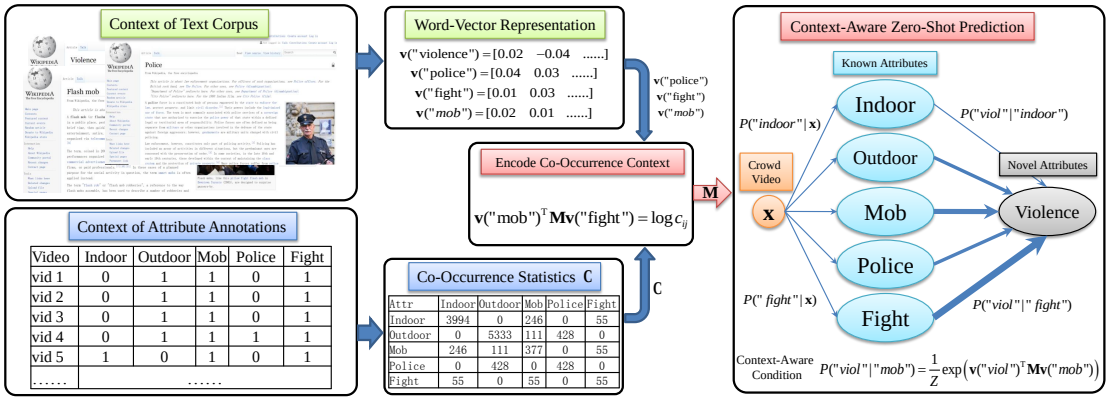


Figure 7.1: Zero-shot crowd behaviour analysis pipeline

To begin with, we present an overview of the pipeline in Figure 7.1. In model training, we learn word-vector representations of training attributes from an external text corpus (context of text corpus), and their visual co-occurrence from the training video annotations (context of attribute annotations). A bilinear mapping $\mathbf{M}$ between pairs of word vectors is trained to predict the log visual co-occurrence statistics $\log c_{ij}$. Visual co-occurrence probabilities can be estimated for any pairs of known or novel (unseen) attributes. To enable the prediction of a novel attribute '*violence*' using the context of known attributes, we first learn a recogniser for each known attribute given its visual features, e.g. $p(\text{`}mob\text{'}|x)$; we then use the trained context model to estimate the conditional probability $P(\text{`}violence\text{'}|\text{`}mob\text{'})$ between novel and known attributes.

Table 7.1: Notation Summary

Notation	Description
N_S; N_T	Number of training/source instances ; testing/target instances
D_x; D_v	Dimension of visual feature; of word-vector embedding
P; Q	Number of training/source attributes ; testing/target attributes
$\mathbf{X} \in \mathbb{R}^{D_x \times N}$; $\mathbf{x}$	Visual feature matrix for N instances; column representing one instance
$\mathbf{Y} \in \{0,1\}^{P \times N}$; $\mathbf{y}$	Binary labels for N instances with P (or Q) labels; column representing one instance
$\mathbf{V} \in \mathbb{R}^{D_v \times P}$; $\mathbf{v}$	Word-Vector embedding for P (or Q) attributes; column representing embedding for one attribute

With this probabilistic structure, we can finally predict the marginalised conditional probability $P(\text{`}violence\text{'}|\mathbf{x})$ to achieve zero-shot crowd behaviour, violence in particular, prediction.

7.1 Probabilistic Zero-Shot Prediction

To account for the multi-label nature of crowd behaviours, we give an overview of the notations used in this chapter in Table 7.1 which is slightly different from Chapter 5 and 6. Formally we have training dataset $T^S = \{\mathbf{X}^S, \mathbf{Y}^S, \mathbf{V}^S\}$ associated with P known attributes and testing dataset $T^T = \{\mathbf{X}^T, \mathbf{Y}^T, \mathbf{V}^T\}$ associated with Q novel/unseen attributes. We denote the visual feature for training and testing videos as $\mathbf{X}^S = [\mathbf{x}_1, \cdots \mathbf{x}_{N^S}] \in \mathbb{R}^{D_x \times N^S}$ and $\mathbf{X}^T = [\mathbf{x}_1, \cdots \mathbf{x}_{N^T}] \in \mathbb{R}^{D_x \times N^T}$, multiple binary labels for training and testing videos as $\mathbf{Y}^S = [\tilde{\mathbf{y}}_1, \cdots \tilde{\mathbf{y}}_{N^S}] \in \{0,1\}^{P \times N^S}$ and $\mathbf{Y}^T = [\mathbf{y^*}_1, \cdots \mathbf{y^*}_{N^T}] \in \{0,1\}^{Q \times N^T}$, and the continuous semantic embedding (word-vector) for training and testing attributes as $\mathbf{V}^S = [\mathbf{v}_1 \cdots \mathbf{v}_P] \in \mathbb{R}^{N^V \times P}$ and $\mathbf{V}^T = [\mathbf{v}_1 \cdots \mathbf{v}_Q] \in \mathbb{R}^{N^V \times Q}$. Note that according to the zero-shot assumption, the training and testing attributes are disjoint i.e. $\forall p \in \{1 \cdots P\}, q \in \{1 \cdots Q\} : \mathbf{v}_p \in \mathbf{V}^S, \mathbf{v}_q \in \mathbf{V}^T, \mathbf{v}_p \neq \mathbf{v}_q$.

To predict novel attributes by reasoning about the relations between known and novel attributes, we formulate this reasoning process as a probabilistic graph (see Figure 7.2).

Given any testing video $\mathbf{x}$, we wish to assign it with one or many of the P known attributes or Q novel attributes. This problem is equivalent to inferring a set of conditional probabilities $p(\mathbf{y}^*|\mathbf{x}) = \{p(y^*_q|\mathbf{x})\}_{q=1\cdots Q}$ and/or $p(\tilde{\mathbf{y}}|\mathbf{x}) = \{p(\tilde{y}_p|\mathbf{x})\}_{p=1\cdots P}$. To achieve this, given the video instance $\mathbf{x}$, we first infer the likelihood of it being one of the P known attributes as $p(\mathbf{y}|\mathbf{x}) = \{p(y_p|\mathbf{x})\}_{p=1\cdots P}$. Then, given the relation between known and novel/known attributes as conditional probability $P(\mathbf{y}^*|\mathbf{y})$ or $P(\tilde{\mathbf{y}}|\mathbf{y})$, we formulate the conditional probability similar to Direct Attribute Prediction (DAP) [19, 143] as follows:

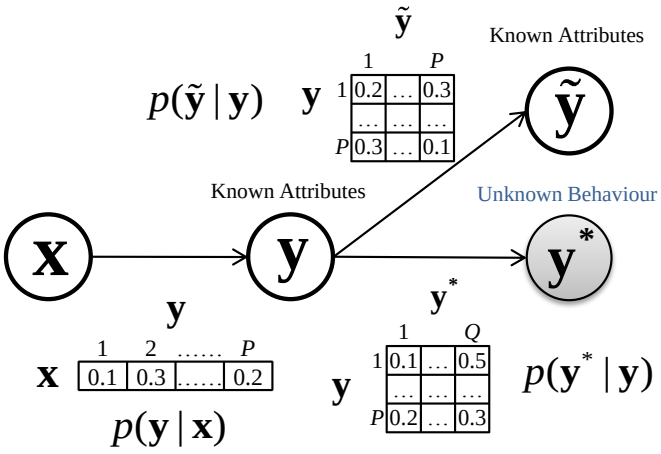


Figure 7.2: A probabilistic graphical representation of a context-aware multi-label zero-shot prediction model.

$$
\begin{aligned}
p(y_q^*|\mathbf{x}) &= \sum_{p=1}^{P} p(y_q^*|y_p)p(y_p|\mathbf{x}) \\
p(\tilde{y}_{\tilde{p}}|\mathbf{x}) &= \sum_{p=1}^{P} p(\tilde{y}_{\tilde{p}}|y_p)p(y_p|\mathbf{x})
\end{aligned}
\tag{7.1}
$$

The zero-shot learning task is to infer the probabilities $\{p(y_q^*|\mathbf{x})\}_{p=1\cdots P}$ for unseen labels $\{y_q^*\}$. We estimate the multinomial conditional probability of known attributes $p(y_p|\mathbf{x})$ based on the output of a probabilistic P-way classifier, e.g. SVM or Softmax Regression with probability output. Then the key to the success of zero-shot prediction is to estimate the known to novel contextual attribute relation as conditional probabilities $\{p(y_q^*|y_p)\}$. We introduce two approaches to estimate this contextual relation.

7.2 Modelling Attribute Relation from Context

In essence, our approach to the prediction of novel attributes depends on the prediction of known attributes and then predicting the novel attributes based on the confidence of each known attribute. The key to the success of this zero-shot prediction is therefore appropriately estimating the conditional probability of novel attribute given known attributes. We first consider a more straightforward way to model this conditional by exploiting the relation encoded by a *text* corpus [143]. We then extend this idea to predict the expected *visual* co-occurrence between novel and known attributes without labelled samples of the novel classes.

7.2.1 Learning Attribute Relatedness from Text Corpora

The first approach builds on semantic word embedding [143]. The semantic embedding represents each English word as a continuous vector $\mathbf{v}$ by training a skip-gram neural network on a large text corpus [24]. The objective of this neural network is to predict optimal adjacent words given the current word, as follows:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0} p(w_{t+j}|w_t) \tag{7.2}$$

The conditional probability is modelled by a softmax function, a normalised probability distribution, based on each word's representation as a continuous vector:

$$p(w_{t+j}|w_t) = \frac{exp(\mathbf{v}_{t+j}^{\top}\mathbf{v}_t)}{\sum_{j=1}^{W} exp(\mathbf{v}_{t+j}^{\top}\mathbf{v}_t)} \tag{7.3}$$

By maximizing the above objective function, the learned word-vectors $\mathbf{V} = \{\mathbf{v}\}$ capture contextual co-occurrence in the text corpora so that frequently co-occurring words result in high cumulative log probability in Eq (7.2). We apply the softmax function to model conditional attribute probability as:

$$p(y_q^*|y_p) = \frac{exp(\frac{1}{\lambda}\mathbf{v}_q^{\top}\mathbf{v}_p^S)}{\sum_{p=1}^{P} exp(\frac{1}{\lambda}\mathbf{v}_q^{\top}\mathbf{v}_p^S)} \tag{7.4}$$

This can be understood intuitively from the following example: An attribute '*Shopping*' has high affinity with attribute '*ShoppingMall*' in word-vector inner product because they co-occur in the text corpus. Our assumption is that the existence of known video attribute '*Shopping*' would support the prediction of unseen attribute '*ShoppingMall*'.

7.2.2 Context Learning from Visual Co-Occurrence

Although attribute relations can be discovered from text context as described above, these relations may *not* ideally suit crowd attribute prediction in videos. For example, the inner product of $vec(\text{'}Indoor\text{'})$ and $vec(\text{'}Outdoor\text{'})$ is 0.7104 which is ranked the 1st w.r.t. 'Indoor' among other 93 attributes in the *WWW* crowd video dataset. As a result, the estimated conditional probability $p(\tilde{y}_{Indoor}|y_{Outdoor})$ is the highest among all $\{p(\tilde{y}_{Indoor}|y_p)\}_{p=1\cdots P}$. However, whilst these two attributes are similar because they occur nearby in the text semantical context, it is counter-intuitive for visual co-occurrence as a video is very unlikely to be *both* indoor and outdoor. Therefore in visual context, their conditional probability should be small rather than large.

To address this problem, instead of directly paramaterising the conditional probability using word-vectors, we use pairs of word vectors to *predict* the actual visual co-occurrence. More precisely, we train a word-vector→co-occurrence predictor based on an auxiliary set of known attributes annotated on videos, for which both word-vectors and annotations are known. We then re-deploy this learned predictor for zero-shot recognition on novel attributes. Formally, given binary multi-label annotations $\mathbf{Y}^S$ on training video data, we define the contextual attribute occurrence as $\mathbf{C} = \mathbf{Y}^S\mathbf{Y}^{S\top}$. The occurrence of j-th attribute in the context of i-th attribute is thus c_{ij} of the $\mathbf{C}$. The prevalence of i-th attribute is defined as $c_i = \sum_j c_{ij}$. The normalised co-occurrence thus defines the conditional probability as:

$$p(\tilde{y}_j|\tilde{y}_i) = \frac{c_{ij}}{c_i} \tag{7.5}$$

The conditional probability can only be estimated based on visual co-occurrence in the case of training attributes with annotations $\mathbf{Y}^S$. To estimate the conditional probability for testing data of novel attributes without annotations $\mathbf{Y}^T$, we consider to *predict* the expected co-occurrence based on a bilinear mapping $\mathbf{M}$ from the pair of word-vectors. Specifically, we approximate the un-normalised co-occurrence as $exp(\mathbf{v}_i^\top \mathbf{M}\mathbf{v}_j) = c_{ij}$. To estimate $\mathbf{M}$, we optimise the regularised linear regression problem as follows:

$$J = \sum_i^P \sum_j^P w(c_{ij}) \left(\mathbf{v}_i^\top \mathbf{M}\mathbf{v}_j - \log c_{ij}\right)^2 + \lambda ||\mathbf{M}||_F^2 \tag{7.6}$$

A weight function $w(c_{ij})$ is applied to the regression loss function above in order to penalise rarely occurring co-occurrence statistics. We choose the weight function according to Pennington et al.[25], which is:

$$w(c_{ij}) = \left(\frac{c_{ij}}{C_{max}}\right)^{(\alpha \cdot \mathbb{1}(c_{ij} \leq C_{max}))} \tag{7.7}$$

where C_{max} is a threshold to control the weight function and $\mathbb{1}$ is an indicator function. This bilinear mapping is related to the model in Pennington et al.[25], but differs in that: (1) the input of the mapping is the word-vector representations $\mathbf{v}$ learned from the skip-gram model [24] in order to generalise to novel attributes where no co-occurrence statistics are available; and (2) the mapping is trained to account for *visual* compatibility, e.g. '*Outdoor*' is unlikely to co-occur with '*Indoor*' in a visual context, although the terms are closely related in their representations learned from the text corpora alone. The bilinear mapping can be seamlessly integrated with the softmax conditional probability as:

$$p(y_q^*|y_p) = \frac{exp(\mathbf{v}_q^\top \mathbf{M}\mathbf{v}_p)}{\sum_p exp(\mathbf{v}_q^\top \mathbf{M}\mathbf{v}_p)} \tag{7.8}$$

Note that by setting $\mathbf{M} = \mathbf{I}$, this conditional probability degenerates to the conventional word-vector based estimation in Eq (7.4). The regression to predict visual co-occurrence from word-vectors (Eq. (7.6)) can be efficiently solved by gradient descent using the following gradient:

$$\nabla\mathbf{M} = \sum_{i=1}^{P}\sum_{j=1}^{P} f(c_{ij})\left(2\mathbf{v}_i\mathbf{v}_i^\top \mathbf{M}\mathbf{v}_j\mathbf{v}_j^\top - 2\log c_{ij}\mathbf{v}_i\mathbf{v}_j^\top\right) + 2\lambda\mathbf{M} \tag{7.9}$$

7.3 Experiments

We evaluate our multi-label crowd behaviour recognition model on the large *WWW* crowd video dataset [10]. We analyse each component's contribution to the overall multi-label ZSL performance. Moreover, we present a proof-of-concept case study for performing transfer zero-shot recognition of violent behaviour in the *Violence Flow* video dataset [56].

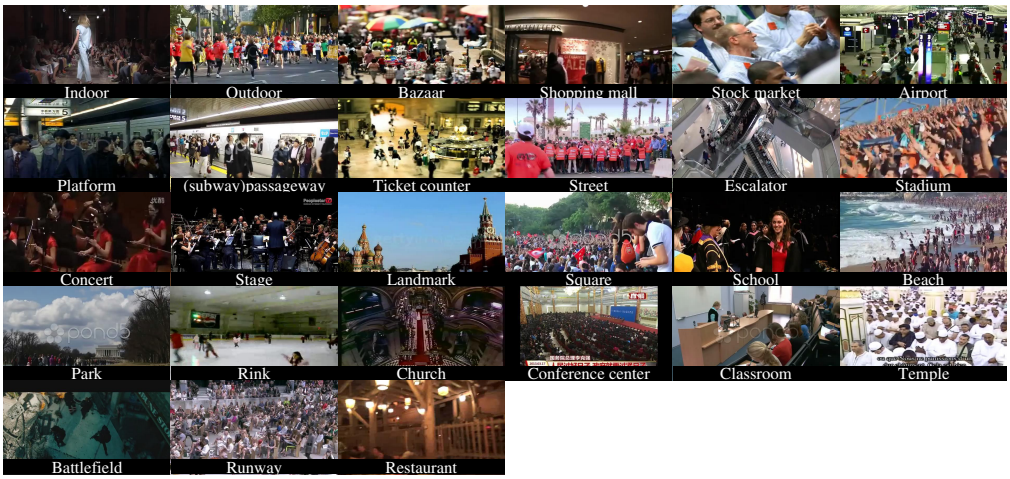
7.3.1 Zero-Shot Multi-Label Behaviour Inference

Dataset

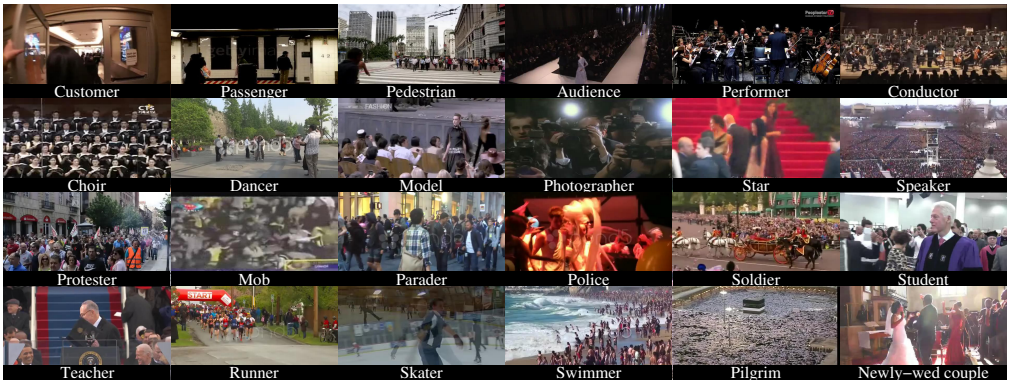
The *WWW* crowd video dataset is specifically proposed for studying scene-independent attribute prediction for crowd scene analysis. It consists of over 10,000 videos collected from online resources of 8,257 unique scenes. The crowd attributes are designed to answer the following questions: 'Where is the crowd', 'Who is in the crowd' and 'Why is the crowd here'. All videos are manually annotated with 94 attributes with 6 positive attributes per video on average. Figure7.3 shows a collection of 94 examples with each example illustrating each attribute in the *WWW* crowd video dataset.

Data Split

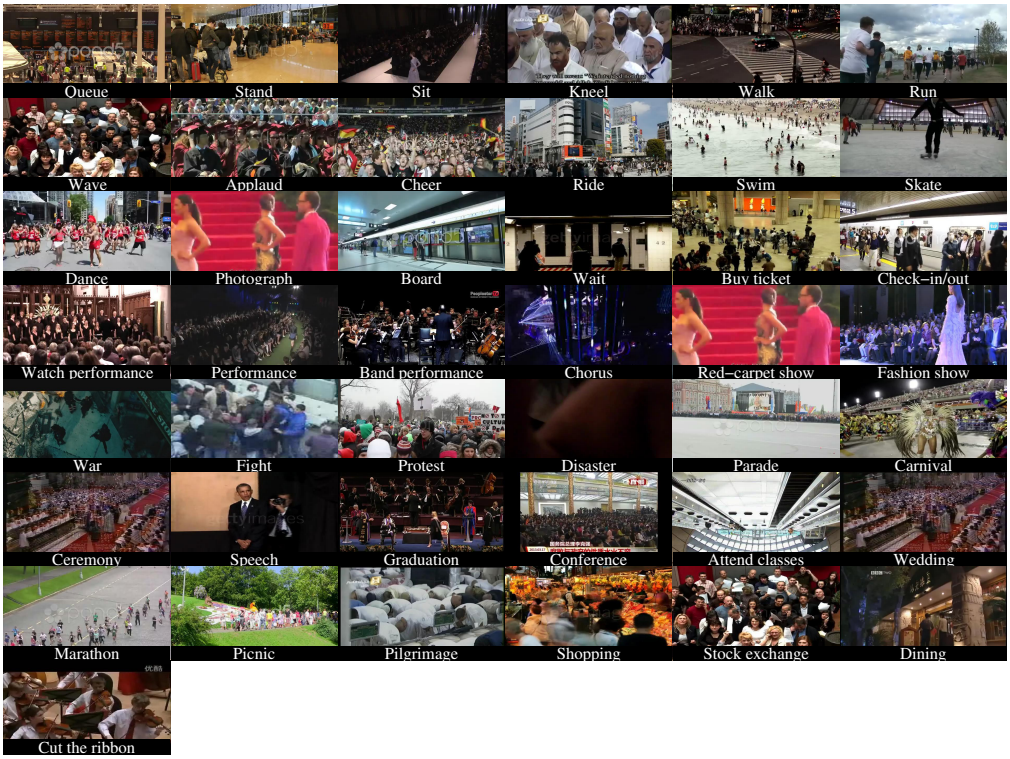
We validated the ability to utilise known attributes for recognising novel attributes in the absence of training samples on the *WWW* dataset. To that end, we divided the 94 attributes into 85 for training (known) and 9 for testing (novel). This was repeated for 50 random splits. In every split, any video which has no positive label from the 9 novel attributes was used for training and the rest for testing. The distributions of the number of multi-attributes (labels) per video over all videos and over the testing videos are shown in Fig 7.4(a-b) respectively. Fig 7.4(c) also shows the distribution of the number of testing videos over the 50 random splits. In most splits, the



(a) 27 attributes by 'Where'

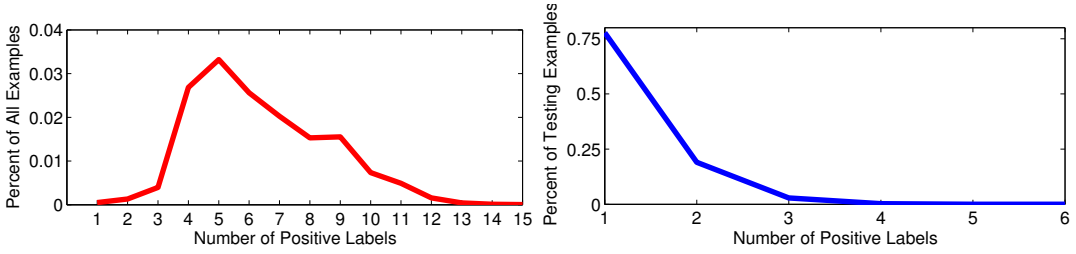


(b) 24 attributes by 'Who'



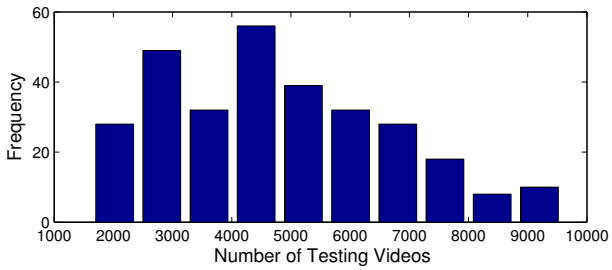
(c) 44 attributes by 'Why'

Figure 7.3: Examples of all attributes in the *WWW* crowd video dataset.



(a) Label distribution over all videos

(b) Label distribution over testing videos



(c) Distribution of testing videos

Figure 7.4: Statistics of the dataset split for our experiments on the *WWW* dataset. (a) and (b): The distributions of multi-label per video over all the videos and over the testing videos respectively. (c): The distribution of the number of testing videos over all 50 random splits.

number of testing videos is in the range of 3,000 to 6,000. The training to testing video number ratio is between 2:1 to 1:1. This low training-testing ratio makes for a challenging zero-shot prediction setting.

Visual Features

Motion information can play an important role in crowd scene analysis. To capture crowd dynamics, we extracted the improved dense trajectory features [15] and performed Fisher vector encoding [92] on these features, generating a $50,688$ dimensional feature vector to represent each video.

Evaluation Metrics

We evaluated the performance of multi-label prediction using five different metrics [53]. These multi-label prediction metrics fall into two groups: Example-based metric and label-based metric. Example-based metrics evaluate the performance per video instance and then average over all instances to give the final metric. Label-based metrics evaluate the performance per label category and return the average over all label categories as the final metric. The five multi-label prediction performance metrics are:

- **AUC** - The Area Under the ROC Curve. AUC evaluates binary classification performance.

It is invariant to the positive/negative ratio of each testing label. Random guess leads to AUC of 0.5. For multi-label prediction, we measure the AUC for each testing label and average the AUC over all 50 splits to yield the AUC per category. The final mean AUC is reported as the mean over all lable categories.

- **Label-based AP** - Label-based Average Precision. We measure the average precision for each attribute as the average fraction of relevant videos ranked higher than a threshold. The random guess baseline for label-based AP is determined by the prevalence of positive videos.

- **Example-based AP** - Example-based Average Precision. We measure the average precision for each video as the average fraction of relevant label prediction ranked higher than a threshold. Example-based AP focuses on the rank of attributes within each instance rather than rank of examples for each label as for label-based AP.

- **Hamming Loss** - Hamming Loss measures the percentage of incorrect predictions from groundtruth labels. Optimal hamming loss is 0, indicating perfect prediction. Due to the nature of hamming loss, the distance of $[000]$ and $[110]$ w.r.t. $[010]$ are equal. Thus it does not differentiate over-estimation from under-estimation. Hamming loss is a label-based metric. The final mean is reported as the average over all instances.

- **Ranking Loss** - Ranking Loss measures, for every instance, the percentage of negative labels ranked higher than positive labels among all possible positive-negative label pairs. Similar to example-based AP, the ranking loss is example-based metric focusing on pushing positive labels ahead of negative labels for each instance.

Both AUC and label-based AP are label-based metrics, whilst exampled-based AP, Hamming Loss and Ranking Loss are example-based metrics. Moreover, as a loss metric, both Hamming Loss and Ranking Loss values are lower the better. In contrary, AUC and AP values are higher the better. In a typical surveillance application of crowd behaviour recognition in videos, we are interested in detecting video instances of a particular attribute that triggers an alarm event, e.g. searching for video instances with the '*fighting*' attribute. In this context, label-based performance metrics such as AUC and Label-based AP are more relevant. Overall, we present model performance evaluated by both types of evaluation metrics.

Comparative Evaluation

In this first experiment, we evaluated zero-shot multi-label prediction on *WWW* crowd video dataset. We compared our context-aware multi-label ZSL models, both purely text-based and visual co-occurrence based, against four contemporary and state-of-the-art zero-shot learning models.

Sate-of-the-art ZSL Models

1. Word-Vector Embedding (**WVE**) [190]: The WVE model constructs a vector representation $\mathbf{z}^{tr} = g(y^{tr})$ for each training instance according to its category name y^{tr} via word-vector embedding $g(\cdot)$ and then learns a support vector regression $f(\cdot)$ to map the visual feature $\mathbf{x}^{tr}$. For testing instance $\mathbf{x}^{te}$, it is first mapped into the semantic embedding space via the regressor $f(\mathbf{x}^{te})$. Novel category $y^{te} \in \mathcal{Y}^{te} = \{1, \cdots Q\}$ is then mapped into the embedding space via $g(y^{te})$. Nearest neighbour matching is applied to match $\vec{x}^{te}$ with category y^* using the L2 distance:

$$y^* = \arg\min_{y_{te} \in \mathcal{Y}^{te}} ||f(\mathbf{x}^{te}) - g(y^{te})||_2^2 \tag{7.10}$$

   We do not assume having access to the whole testing data distribution, so we do not exploit transductive self-training and data augmentation post processing, unlike in the cases of Xu et al.[190] and Alexiou et al.[200].

2. Embarrassingly Simple Zero-Shot Learning (**ESZSL**) [45]: The ESZSL model considers ZSL as training a L2 loss classifier. Specifically, given known categories' binary labels $\mathbf{Y}$ and word-vector embedding $\mathbf{V}^{tr}$, we minimise the L2 classification loss as:

$$\min_{\mathbf{M}} \sum_{i=1}^{N} ||\mathbf{x}_i^\top \mathbf{M} \mathbf{V}^{tr} - \mathbf{y}_i||_2^2 + \Omega(\mathbf{M}; \mathbf{Z}, \mathbf{X}) \tag{7.11}$$

   Novel categories are predicted by:

$$\mathbf{y}^* = \mathbf{x}^{te\top} \mathbf{M} \mathbf{V}^{te} \tag{7.12}$$

3. Extended DAP (**ExDAP**) [158]: ExDAP was specifically proposed for multi-label zero-shot learning [158]. This is an extension of single-label regression models to multi-label.

Specifically, given training instances $\mathbf{x}_i$, associated multiple binary labels $\mathbf{y}_i$, and word-vector embedding of known labels $\mathbf{V}^{tr}$, we minimize the L2 regression loss for learning a regressor $\mathbf{M}$:

$$\min_{\mathbf{M}} \sum_{i=1}^{N} ||\mathbf{x}_i^\top \mathbf{M} - \mathbf{V}^{tr}\mathbf{y}_i||_2^2 + \lambda ||\mathbf{M}||_2^2 \tag{7.13}$$

For zero-shot prediction, we minimize the same loss but w.r.t. the binary label vector $\mathbf{y}$ with L2 regularization:

$$\mathbf{y}^* = \arg\min_{\mathbf{y}^* \in \mathbb{R}} ||\mathbf{x}^{te\top}\mathbf{M} - \mathbf{V}^{te}\mathbf{y}^*||_2^2 + \lambda ||\mathbf{y}^*||_2^2 \tag{7.14}$$

A closed-form solution exists for prediction:

$$\mathbf{y}^* = \left(\mathbf{V}^{te\top}\mathbf{V}^{te} + \lambda \mathbf{I}\right)^{-1} \mathbf{V}^{te\top}\mathbf{x}^{te\top}\mathbf{M} \tag{7.15}$$

4. Direct Multi-Label Prediction (**DMP**) [158]: DMP was proposed to exploit the correlation between testing labels so to benefit the multi-label prediction. It shares the same training procedure with ExDAP in Eq (7.13). For zero-shot prediction, given testing categories $\mathcal{Y}^{te}$ we first synthesise a power-set of all labels $\mathcal{P}(\mathcal{Y}^{te})$. The multi-label prediction $\mathbf{y}^*$ is then determined by nearest neighbour matching of visual instances mapped into word-vector embedding $\mathbf{x}^{te\top}\mathbf{M}$ against the synthesised power-set:

$$\mathbf{y}^* = \arg\min_{\mathbf{y}^* \in \mathcal{P}(\mathcal{Y}^{te})} ||\mathbf{x}^{te\top}\mathbf{M} - \mathbf{V}^{te}\mathbf{y}^*||_2^2 \tag{7.16}$$

Context-Aware Multi-Label ZSL Models

1. Text Context-Aware ZSL (**TexCAZSL**): In our text corpus context-aware model introduced in Section 7.2.1, only word-vectors learned from text corpora [24] are used to model the relation between known and novel attributes $p(\mathbf{y}^*|\mathbf{y})$, as defined by Eq (7.4). We implemented the video instance to known attributes probabilities $p(y_p|\mathbf{x})$ as P linear SVM classifiers with normalised probability outputs [201]. Novel attribute prediction $p(y_q^*|\mathbf{x})$ is computed by marginalising over the known attributes defined by Eq (7.1).

2. Visual Co-occurrence Context-Aware ZSL (**CoCAZSL**): We further implemented a visual co-occurrence context-aware model built on top of the **TexCAZSL** model. This is done

by predicting the expected co-occurrence context using bilinear mapping $\mathbf{M}$, as introduced in Section 7.2.2. The known to novel attribute relation is thus modelled by a weighted inner-product between the word-vectors of known and novel attributes given by Eq (7.8). Novel attribute prediction $p(y_q^*|\mathbf{x})$ is computed in the same way as for **TexCAZSL**, defined by Eq (7.1).

Quantitative Comparison

Table 7.2 shows the comparative results of our models against four state-of-the-art ZSL models and the baseline of 'Random Guess', using all five evaluation metrics. Three observations can be made from these results: (1) all zero-shot learning models can substantially outperform random guessing, suggesting that zero-shot crowd attribute prediction is valid. This should inspire more research into zero-shot crowd behaviour analysis in the future; (2) it is evident that our context-aware models improve on existing ZSL methods when measured by the label-based AUC and AP metrics. As discussed early under evaluation metrics, for typical surveillance tasks, label-based metrics provide a good measurement on detecting novel alarm events in the mist of many other contextual attributes in crowd scenes; and (3) it is also evident that our context-aware models perform comparably to the alternative ZSL models under the example-based evaluation metrics, with the exception that DMP [158] performs extraordinarily well on Hamming Loss but poorly on Ranking Loss. This is due to the direct minimization of Hamming Loss between synthesised power-set and embedded video in DMP. However, since the relative order between attributes are ignored in DMP, low performance in ranking loss as well as other label-based metrics is expected.

Qualitative Analysis

We next give some qualitative examples of zero-shot attribute predictions in Figure 7.5. To get a sense of how well the attributes are detected in the context of label-based AP, we present the AP number with each attribute. Firstly, we give examples of detecting videos matching some randomly chosen attributes (label-centric evaluation). By designating an attribute to detect, we list the crowd videos sorted in the descending order of probability $p(y^*|\mathbf{x})$. In general, we observe good performance in ranking crowd videos according to the attribute to be detected. The false detections are attributed to the extremely ambiguous visual cues. E.g. 3rd video in '*fight*', 5th video in '*police*' and 2nd video in '*parade*' are very hard to interpret.

In addition to detecting each individual attribute, we also present some examples of simultaneously predicting multiple attributes in Figure7.6 (example-centric evaluation). For each video

Table 7.2: Comparison of zero-shot multi-label attribute prediction on the *WWW* crowd video dataset. The ↑ and ↓ symbols indicate whether a metric is higher the better or vice versa.

Feature	**Model**	**Label-Based**		**Example-Based**		
		AUC ↑	**AP ↑**	**AP ↑**	**Hamming Loss ↓**	**Ranking Loss ↓**
-	Random Guess	0.50	0.14	0.31	0.50	-
ITF	WVE[190]	0.65	0.24	0.52	0.45	0.32
ITF	ESZSL[45]	0.63	0.22	0.53	0.46	0.32
ITF	ExDAP[158]	0.62	0.21	0.52	0.45	0.32
ITF	DMP[158]	0.59	0.20	0.45	**0.30**	0.70
ITF	TexCAZSL	0.65	0.24	0.52	0.43	0.32
ITF	CoCAZSL	**0.69**	**0.27**	**0.53**	0.42	**0.31**

we give the the prediction score for all testing attributes as $\{p(y_q^*|\mathbf{x})\}_{q=1\cdots Q}$. For the ease of visualisation, we omit the the attribute with least score. We present the example-based ranking loss number along with each video to give a sense of how the quantitative evaluation metric relates to the qualitative results. In general, ranking loss less than 0.1 would yield very good multi-label prediction as all labels would be placed among the top 3 out of 9 labels to be predicted. Whilst ranking loss around 0.3 (roughly the average performance of our CoCAZSL model, see Table 7.2) would still give reasonable predictions by placing positive labels in the top 5 out of 9.

7.3.2 Transfer Zero-Shot Recognition in Violence Detection

Recognising violence in surveillance scenario has an important role in safety and security [56, 65]. However due to the sparse nature of violent events in day to day surveillance scenes, it is desirable to exploit zero-shot recognition to detect violent events without human annotated training videos. Therefore we explore a proof of concept case study of transfer zero-shot violence detection. We learn to recognise labelled attributes in WWW dataset [10] and then transfer the model to detect violence event in Violence Flow dataset [56]. This is zero-shot because we use no annotated examples of violence to train, and violence does not occur in the label set of WWW. It is contextual because the violence recognition is based on the predicted visual co-occurrence between each known attribute in WWW and the novel violence attribute. For example, '*mob*' and '*police*' attributes known from WWW may support the violence attribute in the new dataset.

Figure 7.5: Illustration of crowd videos ranked in accordance with prediction scores (marginalised conditional probability) w.r.t. each attribute.

Dataset

The Violence Flow dataset [56] was proposed to facilitate the study into classifying violent events in crowded scenes. 246 videos in total are collected from online video repositories, e.g. YouTube, with 3.6 seconds length on average. Half of the 246 video are with positive violence content and the another half are with non-violent crowd contents. We illustrate example frames of both violent and non-violent videos in Figure 7.7

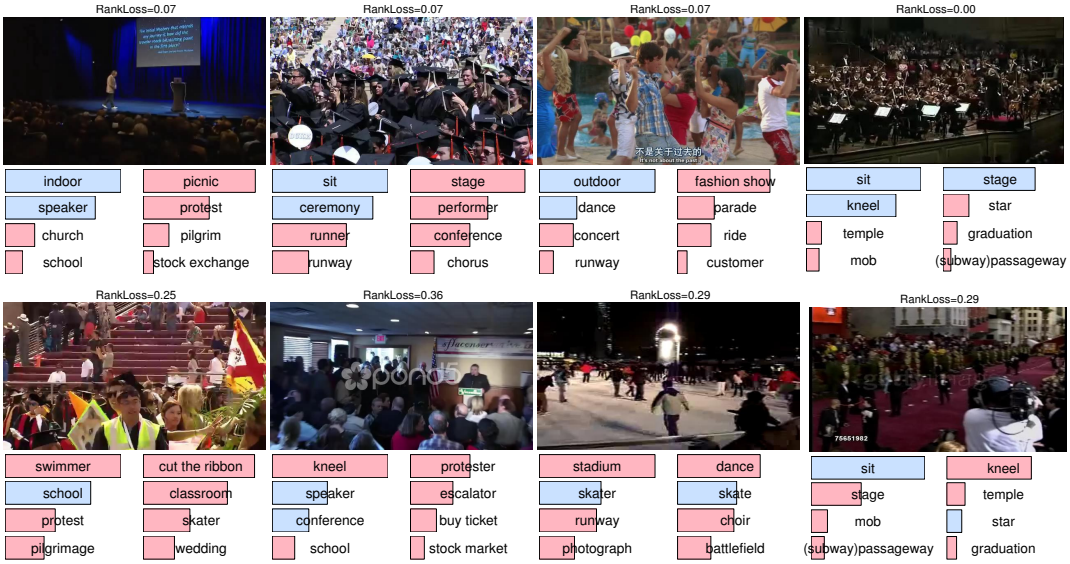


Figure 7.6: Examples of zero-shot multi-label attribute prediction. Bars under each image indicate the normalised score for testing attribtues. Blue and pink bars indicate positive and negative ground-truth labels respectively.

(a) Violent videos

(b) Non-violent videos

Figure 7.7: Example frames of violence flow dataset.

Data Split

A standard fully supervised 5-fold cross validation split was proposed by Hassner et al.[56]. The standard split partitions the whole dataset into 5 splits each of which is evenly divided into positive and negative videos. For each testing split, the other 4 splits are used as the training set and the left-out one is the testing set. Results are reported as both the mean classification

accuracy over 5 splits plus standard deviation and the area under the ROC curve (AUC).

Beyond the standard cross validation split we create a new zero-shot experimental design. Our zero-shot split learns attribute detection models on all 94 attributes from WWW dataset and then tests on the same testing set as the standard 5 splits in Hassner et al.[56]. We note that there are 123 overlapped videos between WWW and Violence Flow. To make fair comparison, we exclude these overlapped videos from constructing the training data for 94 attributes. In this way zero-shot prediction performance can be directly compared with supervised prediction performance using AUC metric. We define the event/attribute to be detected as the word '*violence*'.

Zero-Shot Recognition Models

We explore the transfer zero-shot violence recognition by comparing the same set of zero-shot learning models as in Section 7.3.1: competitors WVE, ESZSL, ExDAP; and our TexCAZSL and CoCAZSL.

Fully Supervised Model

To put zero-shot recognition performance in context, we also report fully supervised models' performance. These models are evaluated on the 5-fold cross-validation split and the average accuracy and AUC are reported. Specifically, we report the best performance of Hassner et al.[56] - linear SVM with VIolent Flows (ViF) descriptor and our fully supervised baseline - linear SVM with Improved Trajectory Feature (ITF).

Results and Analysis

The results of both transfer zero-shot and supervised violence prediction are summarised in Table 7.3. We make the following observations: Our context-aware models perform consistently better than alternative zero-shot models, suggesting that context does facilitate zero-shot recognition. Surprisingly, our zero-shot models moreover perform very competitively compared to the fully supervised models. Our **CoCAZSL** (albeit with better ITF feature) beats the fully supervised Linear SVM with ViF feature in AUC metric (0.87 v.s. 0.85). The context-aware model is also close to the fully supervised model with the same ITF feature (0.87 v.s. 0.99). This is in contrast to the common result in the literature were zero-shot recognition 'works', but does so much worse than fully supervised learning. The promising performance is partly due to modelling the co-occurrence on large known crowd attributes help the correct prediction of known to novel attribute relation prediction.Overall the result shows that by transferring our attribute recognition model trained for a wide set of 94 attributes on a large 10,000 video dataset, it is

possible to perform effective zero-shot recognition of a novel behaviour type in a new dataset.

Table 7.3: Evaluation of violence prediction in Violence Flow dataset: zero-shot versus fully supervised prediction (%).

Model	Split	Feature	Accuracy	AUC
WVE[190]	Zero-Shot	ITF	64.27±5.06	0.64
ESZSL[45]	Zero-Shot	ITF	61.30±8.28	0.62
ExDAP[158]	Zero-Shot	ITF	54.47±7.37	0.52
TexCAZSL	Zero-Shot	ITF	67.07±3.87	0.70
CoCAZSL	Zero-Shot	ITF	**80.52±4.67**	**0.87**
Linear SVM	5-fold CV	ITF	94.72±4.85	0.99
Linear SVM[165]	5-fold CV	ViF	81.30±0.21	0.85

7.3.3 Further Analysis

In this section we provide further analysis on the importance of the visual feature used, and also give more insight into how our contextual zero-shot multi-label prediction works by visualising the learned label-relations.

Feature Analysis

We first evaluate different static and motion features on the standard supervised attribute prediction task. Both hand-crafted and deeply learned features are reported for comparison.

Static Features We report both the hand-crafted and deeply learned static feature from [10] including Static Feature (SFH) and Deeply Learned Static Feature (DLSF). SFH captures general image content by extracting Dense SIFT [179], GIST [178] and HOG [80]. Colour histogram in HSV space is further computed to capture global information and LBP [202] is extracted to quantify local texture. Bag of words encoding is used to create comparable features, leading to a 1536 dimension static feature. DLSF is initialized using a pre-trained model for ImageNet detection task [203] and then fine-tuned on the WWW attribute recognition task with cross-entropy loss.

Motion Features We report both the hand-crafted and deeply learned motion features from Shao et al.[10] including DenseTrack [204], spatio-temporal motion patterns (STMP) [205] and Deeply Learned Motion Feature (DLMF) [10]. Apart from the reported evaluations, we compare

Table 7.4: Comparison between different visual features for attribute prediction.

Alternative Features	Mean AUC
SFH [10]	0.81
DLSF [10]	0.87
DenseTrack [10]	0.63
DLMF [10]	0.68
SFH+DenseTrack [10]	0.82
DLSF+DLMF [10]	0.88
Our Features	
Improved Trajectory Feature (ITF)	0.91

them with the improved trajectory feature (ITF) [15] with fisher vector encoding. Though ITF is constructed in the same way as DenseTrack reported in Shao et al.[10], we make a difference in that the visual codebook is trained on a collection of human action datasets (HMDB51 [4], UCF101 [5], Olympic Sports [7] and CCV [6]).

Analysis Performance on the standard WWW split [10] for static and motion features is reported in Table 7.4. We can clearly observe that the improved trajectory feature is consistently better than all alternative static and motion features. Surprisingly, ITF is even able to beat deep features (DLSF and DLMF). We attribute this to ITF's ability to capture both motion information by motion boundary histogram (MBH) and histogram of flow (HOF) descriptors and texture information by Histogram of Gradient (HOG) descriptor.

More interestingly, we demonstrate that motion feature encoding model (fisher vector) learned from action datasets can benefit the crowd behaviour analysis. Due to the vast availability of action and event datasets and limited crowd behaviour data, a natural extension work is to discover if deep motion model pre-trained on action or event dataset can help crowd analysis.

Qualitative Illustration of Contextual Co-Occurrence Prediction

Recall that the key step in our method's approach to zero-shot prediction is to estimate the visual co-occurrence (between known attributes and held out zero-shot attributes) based on the textually derived word-vectors of each attributes. To illustrate what is learned, we visualise the predicted importance of 94 attributes from WWW in terms of supporting the detection of the held out

attribute '*violence*'. The results are presented as a word cloud in Figure 7.8, where the size of each word/attribute p is proportional to the conditional probability e.g. $p(\text{'}violence\text{'}|y_p)$. As we see from Fig 7.8(a), attribute - '*fight*' is the most prominent attribute supporting the detection of '*violence*'. Besides this, actions like '*street*', '*outdoor*' and '*wave*' all support the existence of '*violence*', while '*disaster*' and '*dining*' among others do not. We also illustrate the support of 'mob' and 'marathon' in Fig 7.8(b) and (c) respectively. All these give us very reasonable importance of known attributes in supporting the recognition of novel attributes.

7.4 Summary

Crowd behaviour analysis has long been a key topic in computer vision research. Supervised approaches have been proposed recently. But these require exhaustively obtaining and annotating examples of each semantic attribute, preventing this strategy from scaling up to ever expanding dataset sizes and variety of attributes. Therefore it is worthwhile to develop recognisers that require little or no annotated training examples for the attribute/event of interest. We address this by proposing a zero-shot learning strategy in which recognisers for novel attributes are built without corresponding training data. This is achieved by learning the recognisers for known labelled attributes. For testing data, the confidence of belonging to known attributes then supports the recognition of novel ones via attribute relation. We propose to model this relation from the co-occurrence context provided by known attributes and word-vector embeddings of the attribute names from text corpora. Experiments on zero-shot multi-label crowd attribute prediction prove the feasibility of zero-shot crowd analysis and demonstrate the effectiveness of learning contextual co-occurrence. A proof of concept case study on transfer zero-shot violence recognition further demonstrates the practical value of our zero-shot learning approach , and its superior efficacy compared to even fully supervised learning approaches.

(a) 'violence' as novel event

(b) 'mob' as novel attribute

(c) 'marathon' as novel attribute

Figure 7.8: Importantce of known attributes w.r.t. novel event/attributes. The fontsize of each attributes is proportional to the conditional probability e.g. $p(\text{`}violence\text{'}|y_p)$.

Chapter 8

Conclusion and Future Work

In this thesis, we address the problem of video behaviour analysis with semantic space discovery. Three specific problems are studied: (1) cross/multi-Scene Understanding and Behaviour Analysis; (2) zero-Shot Action Recognition; and (3) zero-Shot Learning for Multi-Label Crowd Behaviour Analysis. These tasks are non-trivial due to the lack of defined semantic event/scene relatedness, selective sharing information, how to best exploit extra information and multi-label nature. To tackle the event/scene relatedness, we propose to employ geometrical alignment to remove variations caused by viewpoint and further compute semantic relatedness/similarity as a measure of distributions. To selectively share information across semantic scenes, we propose a multi-layer semantic clustering algorithm. For zero-shot action recognition, we introduce three transductive methods which exploit the unlabelled testing data and selectively re-use additional labelled dataset to improve performance. In light of the multi-label nature of crowd behaviour, we propose to learn to predict the co-occurrence between labels to improve the accuracy of recognising novel crowd behaviours. Despite the good performance and insight we made, we believe that out works can serve as good starting points and inspire much more future studies into video behaviour analysis using semantic space.

8.1 Multi-Scene Behaviour Analysis

We introduced a framework in Chapter 4 for synergistically modelling multiple-scene datasets captured by multi-camera surveillance networks. It deals with variable and piece-wise inter-scene relatedness by semantically clustering scenes according to the correspondence of semantic

activities; and selectively shares activities across scenes within clusters. Besides revealing the commonality and uniqueness of each scene, multi-scene profiling further enables typical surveillance tasks of query-by-example, behaviour classification and summarisation to be generalised to multiple scenes. Importantly, by discovering related scenes and shared activities, it is possible to achieve cross-scene query-by-example (in contrast to typical within-scene query), and to annotate behaviour in a novel scene without any labels – which is important for making deployment of surveillance systems scale in practice. Finally, we can provide video summarisation capabilities that uniquely exploit redundancy both within and across scenes by leveraging our multi-scene model.

There are still several limitations to our work which can be addressed in the future:

- In the current framework, scenes that can be grouped together are usually morphologically similar, which means the underlying motion patterns and view angles are essentially similar. More advanced geometrical registration techniques could be applied, including similarity and affine transformations, to allow scenes with more dramatic viewpoint changed to be grouped.

- In multi-scene traffic behaviour analysis, although a semantic space is constructed from traffic data, it is still not fully connected with text knowledge. In another words, every activity learned from scenes are semantic meaningful but not directly related to human language which may further fill the semantic gap.

- In the current framework, motion information is mostly contributed by moving vehicles. A more fine-grained analysis into pedestrian/crowd behaviours under the visual surveillance context would add much more value.

- Incremental learning is of great interest to learning an ever expanding surveillance network. How to extent this model to accommodate to not only new scenes but also automatically adaptively learn new semantic scene clusters remain as open questions.

8.2 Zero-Shot Action Recognition

In Chapter 5 and 6, we investigated *unsupervised* word-vector embedding space representation for zero-shot action recognition for the first time. The fundamental challenge of zero-shot learning is the disjoint training and testing classes, and associated domain-shift. We explored the

impact of four simple but effective strategies to address this: data augmentation, manifold regularization, self-training and hubness correction. Overall we demonstrated that given transductive access to testing data during training stage these strategies are complementary, and together facilitate a highly effective system. To further improve zero-shot nearest neighbour matching, we propose a multi-task embedding strategy. The multi-task approach further mitigates the overfitting to training data and build a lower dimension latent space in which nearest neighbour matching is more meaningful. Finally, we also provide a unique analysis of the inter-class affinity for ZSL, giving insight into why and when ZSL works. This provides for the first time two new capabilities: the ability to predict the efficacy of a given ZSL scenario in advance, and a mechanism to guide the construction of suitable training sets for a desired set of target classes. More importantly, we are inspired to employ a simple re-weighting strategy to selectively use labelled training data. All these together outperform significantly existing methods for zero-shot recognition.

However, the current zero-shot approaches have its own drawbacks. Moreover, the recent development in natural language processing field could inspire a deeper integration between vision and language study and potentially improve zero-shot learning. We list the points which could be addressed in the future:

- First, a fixed pre-trained semantic space, e.g. word-vector, is currently adopted while no contextual information is considered in this procedure. As we have discussed, context may have a huge influence on the semantics, it is desired to consider the semantic space in a context.

- Second, the semantic space is still limited to one or few words (phrase). How to construct semantic space to embed longer and complex textual descriptions is more interesting because this can help us develop visual recognition system for more complex behaviours. The recent sentence vector model [112] proposed an idea to encode a whole sentence or paragraph into a fixed-length vector. The sentence vector could be a good candidate semantic representation for complex event analysis.

- Third, the temporal order of complex action/behaviours are rarely studied to promote performance on zero-shot recognition. Exploiting temporal order in video analysis have been widely conducted in conventional supervised learning [206, 207, 208]. It could be interest-

ing to discover if exploiting the temporal information can help zero-shot action recognition.

- Finally, deep models have shown their superior performance in all aspects of computer vision problems. There are already numerous papers bringing deep models into action recognition [68, 69]. Developing an end-to-end deep zero-shot action recognition model opens a new direction for ZSL study.

8.3 Zero-Shot Crowd Behaviour Analysis

Zero-shot crowd behaviour analysis is studied in Chapter 7. We exploit the manual labelled attributes as an intermediate semantic space and train recognisers for each known attribute. To recognise novel crowd behaviours, we first relate it to the known attributes by word-vector similarity. By predicting novel instance against all known attributes, we can finally predict the score via the discovered relation between known attributes and novel behaviour. Crucially, we note that by modelling the co-occurrence between attributes, we can further improve the performance on recognising novel crowd behaviours.

The future development of zero-shot crowd behaviour analysis is still largely focused on more semantic interpretation, such as semantic crowd video retrieval where query is provided as text rather than video clips [10]. As the existing benchmark, e.g. Violence Flow [56], is still quite limited due to both the dataset size and behaviour type, another important work to be done is collecting interesting but rare crowded surveillance behaviours for case study. At last, all video behaviour analysis problem introduced in this thesis mostly concern with human behaviours. However, they are currently studied separately. An important direction could be jointly exploiting the data and annotations from different sources, e.g. human actions and crowd videos, to mutually benefit each other.

Bibliography

[1] S. Gong and T. Xiang, *Visual analysis of behaviour: from pixels to semantics*. Springer Science & Business Media, 2011.

[2] P. Over, J. Fiscus, G. Sanders, D. Joy, M. Michel, A. F. Smeaton-Alan, and G. Quénot-Georges, "Trecvid 2013–an overview of the goals, tasks, data, evaluation mechanisms, and metrics," 2014.

[3] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: A review," *ACM Computing Surveys*, vol. 43, no. 3, pp. 1–43, 2011.

[4] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *IEEE International Conference on Computer Vision*, 2011, pp. 2556–2563.

[5] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.

[6] Y.-G. Jiang, G. Ye, S.-F. Chang, D. P. W. Ellis, and A. C. Loui, "Consumer video understanding: a benchmark database and an evaluation of human and machine performance." in *ACM International Conference in Multimedia Retrieval*, 2011, p. 29.

[7] J. C. Niebles, C. W. Chen, and L. Fei-Fei, "Modeling temporal structure of decomposable motion segments for activity classification," in *European Conference on Computer Vision*, 2010, pp. 392–405.

[8] J. Li, S. Gong, and T. Xiang, "Learning behavioural context," *International Journal of Computer Vision*, vol. 97, no. 3, pp. 276–304, 2012.

[9] T. Hospedales, S. Gong, and T. Xiang, "Video behaviour mining using a dynamic topic model," *International Journal of Computer Vision*, vol. 98, pp. 303–323, 2012.

[10] J. Shao, K. Kang, C. C. Loy, and X. Wang, "Deeply learned attributes for crowded scene understanding," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4657–4666.

[11] T. Xiang and S. Gong, "Video behavior profiling for anomaly detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 5, pp. 893–908, 2008.

[12] C. Schüldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local SVM approach," in *IEEE International Conference on Pattern Recognition*, 2004, pp. 32–36.

[13] I. Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, no. 2-3, pp. 107–123, 2005.

[14] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[15] H. Wang, D. Oneata, J. Verbeek, and C. Schmid, "A robust and efficient video representation for action recognition," *International Journal of Computer Vision*, vol. 119, no. 3, pp. 219–238, 2016.

[16] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *IEEE International Conference on Computer Vision*, 2005, pp. 1395–1402.

[17] M. Marszalek, I. Laptev, and C. Schmid, "Actions in context," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 2929–2936.

[18] Y.-g. Jiang, Z. Wu, J. Wang, X. Xue, and S.-f. Chang, "Exploiting Feature and Class Relationships in Video Categorization with Regularized Deep Neural Networks," *Arxiv*, pp. 1–22, 2015.

[19] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 951–958.

[20] J. Liu, B. Kuipers, and S. Savarese, "Recognizing human actions by attributes," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 3337–3344.

[21] Y. Fu, T. M. Hospedales, T. Xiang, and S. Gong, "Attribute learning for understanding unstructured social activity," in *European Conference on Computer Vision*, 2012, pp. 530–543.

[22] Y. Fu, T. M. Hospedales, T. Xiang, and S. Gong, "Transductive Multi-view Zero-Shot Learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, pp. 1–17, 2015.

[23] A. Habibian, T. Mensink, and C. G. M. Snoek, "VideoStory: A New Multimedia Embedding for Few-Example Recognition and Translation of Events," in *ACM International Conference on Multimedia*, 2014, pp. 17–26.

[24] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," in *Advances in Neural Information Processing Systems*, 2013, pp. 3111–3119.

[25] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Conference on Empirical Methods in Natural Language Processing*, 2014, pp. 1532–1543.

[26] R. Socher and M. Ganjoo, "Zero-shot learning through cross-modal transfer," in *Advances in Neural Information Processing Systems*, 2013, pp. 935–943.

[27] Y. Fu, T. M. Hospedales, T. Xiang, Z. Fu, and S. Gong, "Transductive multi-view embedding for zero-shot recognition and annotation," in *European Conference on Computer Vision*, 2014, pp. 584–599.

[28] Z. Fu, T. Xiang, E. Kodirov, and S. Gong, "Zero-shot object recognition by semantic manifold distance," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2635–2644.

[29] Z. Akata, S. Reed, D. Walter, H. Lee, and B. Schiele, "Evaluation of output embeddings for fine-grained image classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2927–2936.

[30] X. Wang, X. Ma, and W. E. L. Grimson, "Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 3, pp. 539–555, 2009.

[31] W. Hu, X. Xiao, Z. Fu, D. Xie, T. Tan, and S. J. Maybank, "A system for learning statistical motion patterns." *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1450–1464, 2006.

[32] J. Varadarajan, R. Emonet, and J.-M. Odobez, "A sequential topic model for mining recurrent activities from long term video logs." *International Journal of Computer Vision*, vol. 103, no. 1, pp. 100–126, 2013.

[33] D. Kuettel, M. Breitenstein, L. Van Gool, and V. Ferrari, "What's going on? discovering spatio-temporal dependencies in dynamic scenes," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 1951–1958.

[34] Y. Pritch, A. Rav-Acha, and S. Peleg, "Nonchronological video synopsis and indexing." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 11, pp. 1971–1984, 2008.

[35] X. Wang, K. Ma, G.-W. Ng, and W. Grimson, "Trajectory analysis and semantic region modeling using nonparametric hierarchical bayesian models," *International Journal of Computer Vision*, vol. 95, no. 3, pp. 287–312, 2011.

[36] I. Saleemi, L. Hartung, and M. Shah, "Scene understanding by statistical modeling of motion patterns," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2069–2076.

[37] M. Rodriguez, J. Sivic, I. Laptev, and J.-Y. Audibert, "Data-driven crowd analysis in videos," in *IEEE International Conference on Computer Vision*, 2011, pp. 1235–1242.

[38] J. Shao, C. Loy, and X. Wang, "Scene-independent group profiling in crowd," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2227–2234.

[39] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 3, pp. 453–465, 2014.

[40] S. Khokhar, I. Saleemi, and M. Shah, "Similarity invariant classification of events by kl divergence minimization," in *IEEE International Conference on Computer Vision*, 2011, pp. 1903–1910.

[41] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, 2010.

[42] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov *et al.*, "Devise: A deep visual-semantic embedding model," in *Advances in Neural Information Processing Systems*, 2013, pp. 2121–2129.

[43] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean, "Zero-shot learning by convex combination of semantic embeddings," in *International Conference on Learning Representations*, 2014.

[44] Y.-G. Jiang, J. Liu, A. Roshan Zamir, I. Laptev, M. Piccardi, M. Shah, and R. Sukthankar, "THUMOS challenge: Action recognition with a large number of classes," 2013.

[45] B. Romera-Paredes and P. H. S. Torr, "An embarrassingly simple approach to zero-shot learning," in *International Conference on Machine Learning*, 2015, pp. 2152–2161.

[46] G. Dinu, A. Lazaridou, and M. Baroni, "Improving zero-shot learning by mitigating the hubness problem," in *International Conference on Learning Representations*, 2015.

[47] J. Liu, M. Shah, B. Kuipers, and S. Savarese, "Cross-view action recognition via view knowledge transfer," in *IEEE conference on Computer Vision and Pattern Recognition*. IEEE, 2011, pp. 3209–3216.

[48] J. Zheng and Z. Jiang, "Learning view-invariant sparse representations for cross-view action recognition," in *IEEE International Conference on Computer Vision*, 2013, pp. 3176–3183.

[49] C. Yuan, W. Hu, G. Tian, S. Yang, and H. Wang, "Multi-task sparse learning with beta process prior for action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 423–429.

[50] J. Geng and Z. Miao, "Domain adaptive boosting method and its applications," *Journal of Electronic Imaging*, vol. 24, no. 2, p. 023038, 2015.

[51] S. Mahadevan and S. Chandar, "Reasoning about linguistic regularities in word embeddings using matrix manifolds," *arXiv preprint*, 2015.

[52] J. S. J. Junior, S. Musse, and C. Jung, "Crowd analysis using computer vision techniques," *IEEE Signal Processing Magazine*, vol. 5, no. 27, pp. 66–77, 2010.

[53] M.-L. Zhang and Z.-H. Zhou, "A review on multi-label learning algorithms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 8, pp. 1819–1837, 2014.

[54] N. Ghamrawi and A. McCallum, "Collective multi-label classification," in *ACM International Conference on Information and Knowledge Management*, 2005, pp. 195–200.

[55] X. Li, F. Zhao, and Y. Guo, "Multi-label image classification with a probabilistic label enhancement model," in *The Conference on Uncertainty in Artificial Intelligence*, 2014, pp. 430–439.

[56] T. Hassner, Y. Itcher, and O. Kliper-Gross, "Violent flows: Real-time detection of violent crowd behavior," in *IEEE Conference on Computer Vision and Pattern Recognition Workshop*, 2012, pp. 1–6.

[57] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *The Journal of Machine Learning Research*, vol. 7, pp. 2399–2434, 2006.

[58] M. Sugiyama, S. Nakajima, H. Kashima, P. Von Bünau, and M. Kawanabe, "Direct Importance Estimation with Model Selection and Its Application to Covariate Shift Adaptation," in *Advances in Neural Information Processing Systems*, 2007, pp. 1433–1440.

[59] J. Garcke and T. Vanck, "Importance Weighted Inductive Transfer Learning for Regression," in *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery*, 2014, pp. 466–481.

[60] K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, "When is nearest neighbor meaningful?" *Database Theory*, pp. 217–235, 1999.

[61] T. Evgeniou and M. Pontil, "Regularized multi-task learning," *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 109–117, 2004.

[62] A. Kumar and H. D. III, "Learning Task Grouping and Overlap in Multi-task Learning," *International Conference on Machine Learning*, 2012.

[63] R. Poppe, "A survey on vision-based human action recognition," *Image and Vision Computing*, vol. 28, no. 6, pp. 976–990, 2010.

[64] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 34, no. 3, pp. 334–352, 2004.

[65] S. Gong, C. C. Loy, and T. Xiang, "Security and surveillance," in *Visual Analysis of Humans*, Moeslund, Hilton, Kruger, and Sigal, Eds., 2011, pp. 455–472.

[66] T. Li, H. Chang, M. Wang, B. Ni, R. Hong, and S. Yan, "Crowded scene analysis: A survey," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 3, pp. 367–386, 2015.

[67] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *IEEE conference on Computer Vision and Pattern Recognition*, 2014, pp. 1725–1732.

[68] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems*, 2014, pp. 568–576.

[69] Z. Xu, Y. Yang, and A. G. Hauptmann, "A Discriminative CNN Video Representation for Event Detection," in *IEEE conference on Computer Vision and Pattern Recognition*, 2015, pp. 1798–1807.

[70] B. T. Morris and M. M. Trivedi, "Learning and classification of trajectories in dynamic scenes: A general framework for live video analysis," in *IEEE International Conference on Advanced Video and Signal-Based Surveillance*, 2008, pp. 154–161.

[71] B. T. Morris and M. M. Trivedi, "A survey of vision-based trajectory learning and analysis for surveillance." *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 18, no. 8, pp. 1114–1127, 2008.

[72] B. T. Morris and M. M. Trivedi, "Trajectory learning for activity understanding: Unsupervised, multilevel, and long-term adaptive approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 11, pp. 2287–2301, 2011.

[73] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Computing Surveys*, vol. 38, no. 4, p. 13, 2006.

[74] C. J. Veenman, M. J. Reinders, and E. Backer, "Resolving motion correspondence for densely moving points," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 1, pp. 54–72, 2001.

[75] J. Shi and C. Tomasi, "Good features to track," in *IEEE conference on Computer Vision and Pattern Recognition*, 1994, pp. 593–600.

[76] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[77] D. Russell and S. Gong, "Segmenting highly textured nonstationary background," in *British Machine Vision Conference*, 2007, pp. 1–10.

[78] S. Brutzer, B. Höferlin, and G. Heidemann, "Evaluation of background subtraction techniques for video surveillance," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1937–1944.

[79] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE conference on Computer Vision and Pattern Recognition*, 2001, pp. 511–518.

[80] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2005, pp. 886–893.

[81] S. J. Julier and J. K. Uhlmann, "New extension of the kalman filter to nonlinear systems," in *AeroSense*, 1997, pp. 182–193.

[82] F. Jiang, Y. Wu, and A. K. Katsaggelos, "A dynamic hierarchical clustering method for trajectory-based unusual video event detection," *IEEE Transactions on Image Processing*, vol. 18, no. 4, pp. 907–913, 2009.

[83] C. Liu, "Beyond pixels: Exploring new representations and applications for motion analysis," Ph.D. dissertation, the Massachusetts Institute of Technology, 2009.

[84] T. Brox and J. Malik, "Large displacement optical flow: descriptor matching in variational motion estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 500–513, 2011.

[85] P. Scovanner, S. Ali, and M. Shah, "A 3-dimensional sift descriptor and its application to action recognition," in *ACM Multimedia*, 2007, pp. 357–360.

[86] A. Klaser, M. Marszałek, and C. Schmid, "A spatio-temporal descriptor based on 3d-gradients," in *British Machine Vision Conference*, 2008, pp. 1–10.

[87] H. Wang and C. Schmid, "Action Recognition with Improved Trajectories," in *IEEE International Conference on Computer Vision*, 2013, pp. 3551–3558.

[88] C. Harris and M. Stephens, "A combined corner and edge detector." in *Alvey Vision Conference*, 1988, pp. 1–6.

[89] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *International Journal of Computer Vision*, vol. 103, no. 1, pp. 60–79, 2013.

[90] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[91] I. N. Junejo, O. Javed, and M. Shah, "Multi feature path modeling for video surveillance," in *IEEE International Conference on Pattern Recognition*, 2004, pp. 716–719.

[92] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the Fisher kernel for large-scale image classification," in *European Conference on Computer Vision*, 2010, pp. 143–156.

[93] R. Arandjelovic and A. Zisserman, "All about vlad," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1578–1585.

[94] X. Wang, X. Ma, and E. Grimson, "Unsupervised activity perception by hierarchical bayesian models," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[95] B. Scholkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press. (2002),, 2002.

[96] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[97] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of ACM international conference on Multimedia*, 2014, pp. 675–678.

[98] L. Deng and D. Yu, "Deep learning: Methods and applications," *Foundations and Trends in Signal Processing*, vol. 7, no. 3–4, pp. 197–387, 2014.

[99] J. Deng, A. Berg, S. Satheesh, H. Su, A. Khosla, and L. Fei-Fei, "BWorld Robot Control Software," http://www.image-net.org/challenges/LSVRC/2012/., ILSVRC-2012, [Online; accessed 5-August-2016].

[100] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[101] S. Wu, S. Bondugula, F. Luisier, X. Zhuang, and P. Natarajan, "Zero-Shot Event Detection Using Multi-modal Fusion of Weakly Supervised Concepts," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2665–2672.

[102] R. Emonet, J. Varadarajan, and J.-M. Odobez, "Temporal analysis of motif mixtures using dirichlet processes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 1, pp. 140–156, 2014.

[103] C. Sung, D. Feldman, and D. Rus, "Trajectory clustering for motion prediction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 1547–1552.

[104] J.-G. Lee, J. Han, and K.-Y. Whang, "Trajectory clustering: a partition-and-group framework," in *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, 2007, pp. 593–604.

[105] X. Wang, K. Tieu, and E. Grimson, "Learning semantic scene models by trajectory analysis," in *European Conference on Computer Vision*, 2006, pp. 110–123.

[106] X. L. Xie and G. Beni, "A validity measure for fuzzy clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 8, pp. 841–847, 1991.

[107] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," in *Proceedings of IEEE*, vol. 77, 1989, pp. 257–286.

[108] S. Gong and H. Buxton, "On the visual expectations of moving objects." in *European Conference on Artificial Intelligence*, 1992, pp. 781–784.

[109] T. Hofmann, "Probabilistic latent semantic indexing," in *ACM SIGIR Conference on Research and Development in Information Retrieval*, 1999, pp. 50–57.

[110] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, Mar. 2003.

[111] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, in *Advances in Neural Information Processing Systems*, 2004, pp. 1385–1392.

[112] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents." in *International Conference on Machine Learning*, 2014, pp. 1188–1196.

[113] T. K. Landauer, P. W. Foltz, and D. Laham, "An introduction to latent semantic analysis," *Discourse processes*, vol. 25, no. 2-3, pp. 259–284, 1998.

[114] Q. Wang, J. Xu, H. Li, and N. Craswell, "Regularized latent semantic indexing: A new approach to large-scale topic modeling," *ACM Transactions on Information Systems*, vol. 31, no. 1, p. 5, 2013.

[115] C. Fellbaum, *WordNet*. Wiley Online Library, 1998.

[116] D. Lin, "An information-theoretic definition of similarity," in *International Conference on Machine Learning*, 1998, pp. 296–304.

[117] K. Kim, D. Lee, and I. A. Essa, "Gaussian process regression flow for analysis of motion trajectories." in *IEEE International Conference on Computer Vision*, 2011, pp. 1164–1171.

[118] C. Piciarelli, C. Micheloni, and G. L. Foresti, "Trajectory-based anomalous event detection." *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1544–1554, 2008.

[119] M. Fanaswala and V. Krishnamurthy, "Detection of anomalous trajectory patterns in target tracking via stochastic context-free grammars and reciprocal process models." *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 1, pp. 76–90, 2013.

[120] F.-F. Li and P. Perona, "A bayesian hierarchical model for learning natural scene categories." in *IEEE Conference on Computer Vision and Pattern Recognition*, 2005, pp. 524–531.

[121] L.-J. Li, R. Socher, and F.-F. Li, "Towards total scene understanding: Classification, annotation and segmentation in an automatic framework." in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 2036–2043.

[122] W. Hu, D. Xie, Z. Fu, W. Zeng, and S. Maybank, "Semantic-based surveillance video retrieval," *IEEE Transactions on Image Processing*, vol. 16, no. 4, pp. 1168–1181, Apr. 2007.

[123] T. Xiang and S. Gong, "Activity based surveillance video content modelling," *Pattern Recognition*, vol. 41, no. 7, pp. 2309–2326, 2008.

[124] B. T. Truong and S. Venkatesh, "Video abstraction: A systematic review and classification." *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 3, no. 1, 2007.

[125] S. E. F. de Avila, A. P. B. Lopes, A. da Luz Jr., and A. de Albuquerque Arajo, "Vsumm: A mechanism designed to produce static video summaries and a novel evaluation method." *Pattern Recognition Letters*, vol. 32, no. 1, pp. 56–68, 2011.

[126] Y. Pritch, A. Rav-Acha, A. Gutman, and S. Peleg, "Webcam synopsis: Peeking around the world." in *IEEE International Conference on Computer Vision*, 2007, pp. 1–8.

[127] J.-G. Lou, H. Cai, and J. Li, "A real-time interactive multi-view video system." in *ACM Multimedia*, 2005, pp. 161–170.

[128] Y. Fu, Y. Guo, Y. Zhu, F. Liu, C. Song, and Z.-H. Zhou, "Multi-view video summarization." *IEEE Transactions on Multimedia*, vol. 12, no. 7, pp. 717–729, 2010.

[129] C. d. Leo and B. S. Manjunath, "Multicamera video summarization and anomaly detection from activity motifs," *ACM Transactions on Sensor Networks*, vol. 10, no. 2, pp. 27:1–27:30, Jan. 2014.

[130] J. Varadarajan, R. Emonet, and J.-M. Odobez, "Probabilistic latent sequential motifs: Discovering temporal activity patterns in video scenes." in *British Machine Vision Conference*, 2010, pp. 1–11.

[131] K. Chen, C. C. Loy, S. Gong, and T. Xiang, "Feature mining for localised crowd counting," in *British Machine Vision Conference*, 2012, pp. 1–11.

[132] C. C. Loy, K. Chen, S. Gong, and T. Xiang, "Crowd counting and profiling: Methodology and evaluation," in *Modeling, Simulation and Visual Analysis of Crowds*, Ali, Nishino, Manocha, and Shah, Eds. Springer, December 2013.

[133] S. Ali and M. Shah, "Floor fields for tracking in high density crowd scenes," in *European Conference on Computer Vision*, 2008, pp. 1–14.

[134] E. L. Andrade, S. Blunsden, and R. B. Fisher, "Modelling crowd scenes for event detection," in *IEEE International Conference on Pattern Recognition*, 2006, pp. 175–178.

[135] X. Zhao, D. Gong, and G. Medioni, "Tracking using motion patterns for very crowded scenes," in *European Conference on Computer Vision*, 2012, pp. 315–328.

[136] B. Zhou, X. Tang, and X. Wang, "Coherent filtering: detecting coherent motions from crowd clutters," in *European Conference on Computer Vision*, 2012, pp. 857–871.

[137] H. Larochelle, D. Erhan, and Y. Bengio, "Zero-data learning of new tasks," in *AAAI Conference on Artificial Intelligence*, 2008, pp. 646–651.

[138] M. Palatucci, G. Hinton, D. Pomerleau, and T. M. Mitchell, "Zero-shot learning with semantic output codes," in *Advances in Neural Information Processing Systems*, 2009, pp. 1410–1418.

[139] Y. Fu, T. M. Hospedales, T. Xiang, and S. Gong, "Learning multimodal latent attributes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 2, pp. 303–316, 2014.

[140] F. Zhao, Y. Huang, L. Wang, and T. Tan, "Relevance topic model for unstructured social group activity recognition," in *Advances in Neural Information Processing Systems*, 2013, pp. 2580–2588.

[141] A. Lazaridou, E. Bruni, and M. Baroni, "Is this a wampimuk? cross-modal mapping between distributional semantics and the visual world," in *The Annual Meeting of the Association for Computational Linguistics*, 2014, pp. 1403–1414.

[142] Y. Yang and T. Hospedales, "A unified perspective on multi-domain and multi-task learning," in *International Conference on Learning Representations*, 2015.

[143] C. Gan, M. Lin, Y. Yang, Y. Zhuang, and A. G. Hauptmann, "Exploring Semantic Inter-Class Relationships (SIR) for Zero-Shot Action Recognition," in *AAAI Conference on Artificial Intelligence*, 2015, pp. 3769–3775.

[144] T. Mensink, E. Gavves, and C. G. M. Snoek, "COSTA: Co-occurrence statistics for zero-shot classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2441–2448.

[145] E. Kodirov, T. Xiang, Z. Fu, and S. Gong, "Unsupervised Domain Adaptation for Zero-Shot Learning," in *IEEE International Conference on Computer Vision*, 2015, pp. 2452–2460.

[146] A. Habibian, T. Mensink, and C. G. Snoek, "Composite concept discovery for zero-shot video event detection," in *ACM International Conference on Multimedia Retrieval*, 2014, p. 17.

[147] Q. Zhou, G. Wang, K. Jia, and Q. Zhao, "Learning to share latent tasks for action recognition," in *IEEE International Conference on Computer Vision*, 2013, pp. 2264–2271.

[148] A.-A. Liu, N. Xu, Y.-T. Su, H. Lin, T. Hao, and Z.-X. Yang, "Single/multi-view human action recognition via regularized multi-task learning," *Neurocomputing*, pp. 544–553, 2015.

[149] B. Mahasseni and S. Todorovic, "Latent multitask learning for view-invariant action recognition," in *IEEE International Conference on Computer Vision*, 2013, pp. 3128–3135.

[150] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1521–1528.

[151] J. Huang, A. J. Smola, A. Gretton, K. M. Borgwardt, and B. Schölkopf, "Correcting Sample Selection Bias by Unlabeled Data," in *Advances in Neural Information Processing Systems*, 2007, pp. 601–608.

[152] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *European Conference on Computer Vision*, 2010, pp. 213–226.

[153] D. Pardoe and P. Stone, "Boosting for Regression Transfer," in *International Conference on Machine Learning*, 2010, pp. 863–870.

[154] A. Gretton, K. M. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola, "A kernel method for the two-sample-problem," in *Advances in Neural Information Processing Systems*, 2006, pp. 513–520.

[155] M. Baktashmotlagh, M. Harandi, B. Lovell, and M. Salzmann, "Unsupervised domain adaptation by domain invariant projection," in *IEEE International Conference on Computer Vision*, 2013, pp. 769–776.

[156] M. Radovanovic, A. Nanopoulos, and M. Ivanovic, "Hubs in space: Popular nearest neighbors in high-dimensional data," *Journal of Machine Learning Research*, vol. 11, pp. 2487–2531, 2010.

[157] N. Tomasev, "The role of hubness in high-dimensional data analysis," *Informatica*, vol. 38, no. 4, p. 387, 2014.

[158] Y. Fu, Y. Yang, T. M. Hospedales, T. Xiang, and S. Gong, "Transductive Multi-Label Zero-shot Learning," in *British Machine Vision Conference*, 2014.

[159] M.-L. Zhang and Z.-H. Zhou, "Ml-knn: A lazy learning approach to multi-label learning," *Pattern recognition*, vol. 40, no. 7, pp. 2038–2048, 2007.

[160] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown, "Learning multi-label scene classification," *Pattern recognition*, vol. 37, no. 9, pp. 1757–1771, 2004.

[161] G. Tsoumakas and I. Vlahavas, "Random k-labelsets: An ensemble method for multilabel classification," in *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery*, 2007, pp. 406–417.

[162] J. Fürnkranz, E. Hüllermeier, E. L. Mencía, and K. Brinker, "Multilabel classification via calibrated label ranking," *Machine learning*, vol. 73, no. 2, pp. 133–153, 2008.

[163] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, T. Mei, and H.-J. Zhang, "Correlative multi-label video annotation," in *ACM International Conference on Multimedia*, 2007, pp. 17–26.

[164] N. Ueda and K. Saito, "Parametric mixture models for multi-labeled text," in *Advances in Neural Information Processing Systems*, 2002, pp. 721–728.

[165] B. Hariharan, S. Vishwanathan, and M. Varma, "Efficient max-margin multi-label classification with applications to zero-shot learning," *Machine learning*, vol. 88, no. 1-2, pp. 127–155, 2012.

[166] C. C. Loy, T. Xiang, and S. Gong, "Incremental activity modeling in multiple disjoint cameras," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, pp. 1799–1813, 2012.

[167] J. Prokaj, X. Zhao, and G. G. Medioni, "Tracking many vehicles in wide area aerial surveillance." in *IEEE International Conference on Computer Vision and Pattern Recognition Workshops*, 2012, pp. 37–43.

[168] D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized lline or its caricature," *Cartographica: The International Journal for Geographic Information and Geovisualization*, vol. 10, no. 2, pp. 112–122, Oct. 1973.

[169] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering." in *Advances in Neural Information Processing Systems*, 2004, pp. 1601–1608.

[170] J. Hershey and P. Olsen, "Approximating the kullback leibler divergence between gaussian mixture models," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2007, pp. 317–320.

[171] H. Kuhn, "The Hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[172] A. Goshtasby, "Image registration by local approximation methods." *Image Vision Comput.*, vol. 6, no. 4, pp. 255–261, 1988.

[173] J. Nocedal and S. Wright, *Numerical optimization*, 2nd ed. Springer-Verlag, 2006.

[174] Federal Highway Administration, "Next generation simulation (ngsim) dataset," http://ops.fhwa.dot.gov/trafficanalysistools/ngsim.htm, 2007, [Online; accessed 22-August-2016].

[175] J. Varadarajan and J. Odobez, "Topic models for scene analysis and abnormality detection," in *IEEE International Conference on Computer Vision, Computer Vision Workshops*, 2009, pp. 1338–1345.

[176] J. Xiao, K. A. Ehinger, J. Hays, A. Torralba, and A. Oliva, "SUN database: Exploring a large collection of scene categories," *International Journal of Computer Vision*, vol. 119, no. 1, pp. 3–22, 2016. [Online]. Available: http://dx.doi.org/10.1007/s11263-014-0748-y

[177] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using places database," in *Advances in neural information processing systems*, 2014, pp. 487–495.

[178] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[179] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2006, pp. 2169–2178.

[180] J. Zheng, Z. Jiang, P. J. Phillips, and R. Chellappa, "Cross-view action recognition via a transferable dictionary pair," in *British Machine Vision Conference*, 2012, pp. 1–11.

[181] T. F. Gonzalez, "Clustering to minimize the maximum intercluster distance." *Theoretical Computer Science*, vol. 38, pp. 293–306, 1985.

[182] Hochbaum and Shmoys, "A best possible heuristic for the k-center problem," *Mathematics of Operations Research*, vol. 10, no. 2, pp. 180–184, 1985.

[183] J. Zheng and Z. Jiang, "Learning view-invariant sparse representations for cross-view action recognition," in *IEEE International Conference on Computer Vision*, 2013, pp. 3176–3183.

[184] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[185] C.-W. Ngo, Y.-F. Ma, and H. Zhang, "Video summarization and scene detection by graph modeling." *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 2, pp. 296–305, 2005.

[186] Y.-F. Ma, X.-S. Hua, L. Lu, and H. Zhang, "A generic framework of user attention model and its application in video summarization." *IEEE Transactions on Multimedia*, vol. 7, no. 5, pp. 907–919, 2005.

[187] W. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical Association*, vol. 66, no. 336, pp. 846–850, 1971.

[188] J. Mitchell and M. Lapata, "Vector-based Models of Semantic Composition," *Computational Linguistics*, vol. 8, no. June, pp. 236–244, 2008.

[189] D. Milajevs, D. Kartsaklis, M. Sadrzadeh, and M. Purver, "Evaluating neural word representations in tensor-based compositional settings," in *Conference on Empirical Methods in Natural Language Processing*, 2014, pp. 708–719.

[190] X. Xu, H. Timothy, and S. Gong, "Semantic embedding space for zero shot action recognition," in *IEEE Conference on Image Processing*, 2015, pp. 63–67.

[191] M. Rohrbach, S. Ebert, and B. Schiele, "Transfer learning in a transductive setting," in *NIPS*, 2013.

[192] M. Rohrbach, A. Rohrbach, M. Regneri, S. Amin, M. Andriluka, M. Pinkal, and B. Schiele, "Recognizing fine-grained and composite activities using hand-centric features and script data," *IJCV*, 2016.

[193] D. Zhou, O. Bousquet, and J. Weston, "Learning with local and global consistency," in *NIPS*, 2004.

[194] L. V. D. Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[195] J. Zheng and Z. Jiang, "Submodular Attribute Selection for Action Recognition in Video," in *Advances in Neural Information Processing Systems*, 2014, pp. 1–9.

[196] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[197] H. Drucker, "Improving regressors using boosting techniques," in *International Conference on Machine Learning*, 1997, pp. 107–115.

[198] L. Van Der Maaten, "Accelerating t-sne using tree-based algorithms," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3221–3245, 2014.

[199] C. Gan, N. Wang, Y. Yang, D.-Y. Yeung, and A. G. Hauptmann, "Devnet: A deep event network for multimedia event detection and evidence recounting," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2568–2577.

[200] I. Alexiou, T. Xiang, and S. Gong, "Exploring synonyms as context in zero-shot action recognition," in *IEEE International Conference on Image Processing*, 2016, pp. 4190–4194.

[201] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, 2011, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[202] G. Zhao, T. Ahonen, J. Matas, and M. Pietikäinen, "Rotation-invariant image and video description with local binary pattern features," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1465–1477, 2012.

[203] W. Ouyang, P. Luo, X. Zeng, S. Qiu, Y. Tian, H. Li, S. Yang, Z. Wang, Y. Xiong, C. Qian *et al.*, "Deepid-net: multi-stage and deformable deep convolutional neural networks for object detection," *arXiv preprint arXiv:1409.3505*, 2014.

[204] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *IEEE conference on Computer Vision and Pattern Recognition*, 2011, pp. 3169–3176.

[205] L. Kratz and K. Nishino, "Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models." in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1446–1453.

[206] B. Fernando, E. Gavves, J. M. Oramas, A. Ghodrati, and T. Tuytelaars, "Modeling video evolution for action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5378–5387.

[207] K. Tang, L. Fei-Fei, and D. Koller, "Learning latent temporal structure for complex event detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1250–1257.

[208] S. Bhattacharya, M. M. Kalayeh, R. Sukthankar, and M. Shah, "Recognition of complex events: Exploiting temporal dynamics between underlying concepts," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2235–2242.